THE UNIVERSITY
of EDINBURGH

Developing a framework for semi-automated rule-based modelling for neuroscience research

Emilia M. Wysocka

Doctor of Philosophy

Institute for Adaptive and Neural Computation
School of Informatics
University of Edinburgh

2018

Abstract

Dynamic modelling has significantly improved our understanding of the complex molecular mechanisms underpinning neurobiological processes. The detailed mechanistic insights these models offer depend on the availability of a diverse range of experimental observations. Despite the huge increase in biomolecular data generation from novel high-throughput technologies and extensive research in bioinformatics and dynamical modelling, efficient creation of accurate dynamical models remains highly challenging. To study this problem, three perspectives are considered: comparison of modelling methods, prioritisation of results and analysis of primary data sets. Firstly, I compare two models of the DARPP-32 signalling network: a classically defined model with ordinary differential equations (ODE) and its equivalent, defined using a novel rule-based (RB) paradigm. The RB model recapitulates the results of the ODE model, but offers a more expressive and flexible syntax that can efficiently handle the "combinatorial complexity" commonly found in signalling networks, and allows ready access to fine-grain details of the emerging system. RB modelling is particularly well suited to encoding protein-centred features such as domain information and post-translational modification sites. Secondly, I propose a new pipeline for prioritisation of molecular species that arise during model simulation using a recently developed algorithm based on multivariate mutual information (CorEx) coupled with global sensitivity analysis (GSA) using the RKappa package. To efficiently evaluate the importance of parameters, Hilber-Schmidt Independence Criterion (HSIC)-based indices are aggregated into a weighted network that allows compact analysis of the model across conditions. Finally, I describe an approach for the development of disease-specific dynamical models using genes known to be associated with Attention Deficit Hyperactivity Disorder (ADHD) as an exemplar. Candidate disease genes are mapped to a selection of datasets that are potentially relevant to the modelling process (e.g. interactions between proteins and domains, protein-domain and kinase-substrates mappings) and these are jointly analysed using network clustering and pathway enrichment analyses to evaluate their coverage and utility in developing rule-based models.

Lay Summary

The activity of a cell and its function is driven by interacting molecules, in particular proteins. As proteins can have functionally different states and bind more than one partner, a possible number of ways and outcomes of such interactions can be difficult to trace. The unfolding of exact mechanics behind these interactions becomes particularly important when protein functions are disrupted leading to disease. Their intricate character greatly hampers understanding of exact reasons and mechanisms of such disruptions that to be effectively counteracted require accurately designed medications. Details of such interactions have been traditionally studied with computational models formulated as mathematical equations, in particular, ordinary differential equations (ODEs). These models attempt to reproduce experimental observations by integrating existing knowledge into one controllable and explicit representation. However, not all molecular processes are representable with this traditional method. Moreover, gathering detailed information to construct such models is a slow and laborious process what limits the number of studied molecules and research goals. Possibility to circumvent these issues with a relatively novel rule-based (RB) modelling approach is examined in this thesis. This is attempted in three areas. Firstly, ODE and RB modelling approaches are compared based on one model represented with the two methods to identify differences and conditions under which the RB model is advantageous. Secondly, a pipeline for RB model analysis is proposed to automatically identify which molecules and reactions become important dependent on conditions, such as a disrupted versus an undisrupted state. Lastly, to learn if model building can be accelerated and simplified, large scale data acquired with cost-efficient high-through methods, typically unused in ODE-type of modelling, are collected from public databases. Among these data sets are pairs of interacting proteins and their functional segments (domains). Scope and detail of these datasets are examined by asking a question what molecular mechanisms underlie Attention Deficit Hyperactivity Disorder (ADHD).

Acknowledgements

Foremost, I am profoundly grateful to Ian Simpson for his careful and patient supervision of my research pursuit. Without his guidance, enthusiasm, continuous support, and sharing his greatly versatile experience and knowledge, this thesis would not be here.

My sincere thanks to Matt Page and James Snowden, my external supervisors from UCB Pharma, who gave me the perspective of pharmaceutical application of this project and guided me over these years.

I would like to thank to my second supervisor, Douglas Armstrong, for advice, insightful remarks, and discouragement of favouring a single protein.

I would like to thank to Oksana Sorokina, a member of the thesis committee and a rule-based practitioner whose advice I could always seek without hesitation.

Many thanks to Anatoly Sorokin for his help and clarifications on global sensitivity analysis for rule-based models.

I am very thankful to Greg Ver Steeg for enlightening me in the inner workings of CorEx and Sahil Garg for the idea of using this particular method.

Many thanks to Xin He for his guidance and assistance in application of the topONTO package.

I am grateful for taking part in inspiring Rule-Based Modelling Group meetings with Vincent Danos, Ricardo Honorato Zimmer, Tobias Heindel and many others, with whom I could discuss the ways of Kappa.

I thank to Katharina Heil and David Sterratt, who were my very first co-authors. Thank you for this experience. I learned plenty.

I would like to thank to Ian's Statistical Bioinformatics Group: Ian, Xin, Maciej Pajak, Alba Crespi, Owen Dando, and Sam Heron, for great conferences together, invading space and saving the world from evil Ancient Ones during memorable Board Gaming Nights.

I would like to thank to University of Edinburgh and UCB Pharma for founding this exciting project of multiple forking paths that I could study with enchantment.

I thank to my dear office mates of IF2.53: Katharina, Alba, Xin, Nathalie Dupuy, Martino Sorbaro, Jin Lee Hu, and Joseph Cronin. Thanks for being a cheerful troupe.

Finally, I thank to Przemysław Sanecki, for his love songs from the saint mountains of Silesia and telling me that this thesis is a good read. This would not happen without you.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Emilia M. Wysocka*)

Contents

Chapter 1

Introduction

1.1 Cell signalling

Cells of multicellular and unicellular organisms developed mechanisms to perceive and respond to environment with a range of behaviours. In multicellular organisms, their behaviours vary from differentiaton, apoptosis, growth or survival, and depend on the cell history and its current state. Cell signalling is an overarching term encompassing mechanisms of cell communication and its outcome determining cell response [1]. It can be characterised with such concepts as input fidelity, output specificity, signal amplification, sensitivity and diversity of response, and flexibility of reaction, features commonly associated with electrical engineering [2]. Such complex properties of molecular signalling emerge from characteristics of proteins, major players of any subcellular process, and biochemical events they are involved in.

1.1.1 Proteins as key players of molecular signalling

Proteins are characterised with amino acid sequences, protein domain compositions, protein 3D structures and functions. Based on similarity in these features, proteins are classified into *families* and define their properties. Family definitions classify and annotate newly sequenced proteins with functions [3]. Understanding protein functions is also studied through protein binding characteristics and interactions. Binding properties of proteins are often determined by their functional subunits, such as *domains*, *repeats* and *motifs*.

Domains are autonomous and stably folded segments of amino acid chains that can mediate protein binding and catalytic events [1]. It is estimated that around 65% of eukaryotic proteins are composed of more than one domain

[4]. Compositions of domains in proteins, termed *domain architecture*, can define their unique identities. As domains differ with respect to mediated functional roles, rearrangements of their compositions underlie evolution of novel classes of functionally sophisticated proteins [5–7]. Elementary events shaping new domain architectures are deletions and insertions, repetitions and exchanges. They can occur through gene fusion/fission identified as an important factor of multi-domain protein evolution in bacteria [8]. The evolutionary conservation of domains is also connected with preservation of binding properties between domains. Domain interactions are divided into homo- or hetero-domain that occur between the same and different domains, respectively. Domain interactions are also categorised as inter- and intra-chain interactions. Inter-chain interactions take place between two different polypeptide chains whereas intra-chain interactions, between domains composing the same chain [7].

Another type of functionally separate protein modules are short linear motifs (SLiMs). SLiMs are encoded by peptides of 3 to 10 amino acids, typically located in intrinsically disordered proteins that do not form a stable 3D structure [9]. SLiMs were found to perform multiple diverse functions, such as cell regulation, cooperative formation of protein complexes [10] and internalization and trafficking of membrane receptors [11]. SLiM-mediated interactions have relatively low binding affinity what characterise them as transient, conditional and tunable, ideal to mediate cell signalling [12].

Next to modularity of proteins, another biochemical mechanism that diversifies protein function and activity is their modulation by post-translational modifications (PTMs) [13]. PTMs are covalent protein modifications that occur after protein synthesis. Though there are multiple types of PTMs, phosphorylation is the most important and abundant form [14]. Phosphorylation is an enzymatic reaction of phosphate group transfer from adenosine triphosphate (ATP) to particular amino acids of the substrate protein, such as Serine, Threonine or Tyrosine. Phosphorylation is catalysed by protein kinases. By inducing conformational changes that exposes or hides protein active sites, phosphorylation activates or inhibits activity of proteins, such as enzymes and receptors [15, 16]. Phosphorylation can also trigger translocation of a protein to a different cell compartment [17]. A phosphorylated protein site is recognised and bound by proteins with phospho-binding domains [5]. Reverse reaction of detachment of the phosphate group from substrate proteins is catalysed by

protein phosphatase enzymes [1].

Proteins involved in signalling often are regulated by multiple phosphorylation sites, such as the canonical example of epidermal growth factor receptor (EGFR) that is composed of more than 9 Tyrosine phosphorylation sites, alongside multiple protein domains [18]. EGFR is one of a large family of transmembrane receptor Tyrosine kinases [19]. It is an important target of multiple extracellular signals. When activated, it dimerises and a particular subset of phosphorylation sites become autophosphorylated. These phosphorylated sites bind to a number of protein complexes that trigger distinctive cell behaviours, such as cell cycle or transcription [18].

A variety of mechanisms are connected to the ability of proteins to bind multiple partners. For instance, proteins can form heterogeneous complexes by docking multiple partners. Important for signalling are special types of protein complexes called *scaffold proteins* [20]. Composed of multiple protein domains and motifs, protein scaffolds have diverse interaction interfaces. They bind several signalling proteins at a time whereby they are involved in multiple parallel reactions. Protein scaffolds form functional modules with other signalling proteins locating signalling units in close milieu. This mechanism secures selectivity and efficiency of the cellular signal [20, 21]. Scaffold proteins can distally control activation of their binding partners through *allosteric regulation* or themselves be subjected to such regulation [20]. Allosteric regulation describes mechanisms where a molecule can regulate enzyme activity by binding to its regulatory site that affects conformation of the enzyme's active site. Alteration of the active site either enables or blocks binding and activation of the enzyme target [22]. If a protein has multiple binding sites [23], allosteric regulation can lead to *cooperative binding*. This mechanism occurs when binding of one molecule affects the strength of binding (affinity) of other molecules [23].

Another mechanism grounded in various levels of promiscuity in interactions between signalling proteins and small molecules is *competition*. Competition is an important mechanism in cell signalling. It can be observed on examples of multiple substrates activated by a single promiscuous kinase, or a phosphatase [24, 25].

A series of consecutive reactions between molecules that produce a certain outcome are known as *molecular pathways*. Phosphorylation cascades are

characteristic pathways in cell signalling, where an activated substrate becomes kinase of the subsequent reaction. A canonical example of phosphorylation cascade is the mitogen-activated protein kinase cascade [26].

Triggered by multiple incoming signals, molecular pathways do not act alone but interact with each other forming *cross-talks* between pathways. In signalling networks, such inter-pathway interactions are commonly founded by promiscuous kinases that targets substrates of multiple pathways [25]. These multiple interacting pathways form complex *molecular networks* [27].

Cross-talks between pathways contribute to diversity of dynamic response and therefore, allow the cell to discriminate between specific combinations of input signals [28]. These complex interactions are regulated and controlled to maintain specificity and fidelity between the input signal and the output response. Among highly sophisticated dynamic non-linear responses are ultrasensitivity, multistability, and oscillations [29, 30]. For instance the mitogen-activated protein kinase cascade was found to produce ultrasenstivie dynamic response, characterised by a sudden switch from one state to another [26]. However, there are other mechanisms that are known to trigger ultrasensitivity, such as substrate competition [24, 25]. It has been observed that complex dynamic responses can be attributed to particular topologies of circuits embedded within biological networks that induce control over signalling systems.

1.1.2 Circuits of signalling control

In analogy to control systems studied in engineering, it has been observed that in biological networks, such as transcription and signalling networks, sequences of interactions between genes or proteins form regulatory circuits that control signals. They are composed of activation and inhibition interactions between input (source) and output (target) signals. These regulatory circuits are called *network motifs* [31]. Presence of such motifs in biological networks is manifestation of signal-processing functions existing within complex networks of interactions [31]. Elementary regulation types found among network motifs are: *simple regulation* when activation between two different genes or proteins occurs directly without intermediate steps, *negative* or *positive autoregulation* when a gene or protein inhibits or activates itself, respectively [31]. These simple regulation motifs form composite circuits divided into *feedback*

loops and *feedforward loops* [27, 32]. A feedback loop is a control circuit where the output signal returns as an input to regulate the signal source after one or more steps. Feedback loops are divided into positive and negative depending on whether the returning input has activating or inhibiting effect, respectively. In general, a *positive feedback loop* amplifies the signal, whereas *the negative feedback loop* attenuates it. The other type of control circuit is a feedforward loop. In this circuit, a single molecule inhibits or activates two others thereby the signal is forked into two routes that meets again after a few steps on the same target protein or gene. One of these two routes is usually a direct control of the target by the signal source and the other route is composed of a chain of connected interactors, either inhibiting or activating one another. Feedforward loops are classified into two general types. To explain these classification, let us assume that inhibiting actions per interactor are denoted with a minus sign and positive ones with a plus sign. If result of multiplication of signs in indirect route agrees with the sign on the direct route, then this feedforward loop is called *coherent*, otherwise it is *incoherent* [33]. Identification of such motifs in complex signalling networks characterises dynamics of involved molecules and response of the system under study in such terms as robustness to perturbations, adaptation and threshold response [31, 34].

1.1.3 Neuronal signalling

Groups of communicating neurons form neural circuits that pass electrical signals through action potentials [35]. The signal between two communicating neurons is passed biochemically through *synapses* during the process of *synaptic transmission*. Repeated interactions of neurons cause strengthening or weakening of neuronal synapses having impact on efficiency of signal transmission. These action-dependent modifications of synapses define *synaptic plasticity* that underlies foundation of learning and memory [36]. Synaptic transmission can be modified for short period of time, from miliseconds to minutes, or for much longer, lasting from minutes to hours and days [37]. Prolonged increase in strength and efficiency of synaptic transmission is termed long-term potentiation (LTP) and opposite process is called long-term depression (LTD) [38]. Among others (e.g. metaplasticity, spike timing dependent plasticity), these are the most intensively studied forms of activity-dependent synaptic plasticities [37]. LTP and LTD were first studied in hippocampal neu-

rons where they are conditioned by the amount of calcium ions (Ca^{2+}) influx through N-methyl-D-aspartate receptor (NMDAR) activated by glutamatergic neurotransmitters. High levels of Ca^{2+} indirectly activates Ca^{2+}/calmodulin-dependent protein kinase II (CaMKII) that results with LTP. Moderate levels of Ca^{2+} activate protein phosphatase 3/calcineurin (PP2B) producing LTD [38]. These mechanisms triggering either LTD or LTP are characteristic for hippocampal neurons. In other brain regions, activity-based synaptic plasticity can be produced by different mechanisms. There are multiple factors that contribute to variety of mechanisms underlying synaptic plasticity in other brain regions, such as different types of neurotransmitters, receptors and spatio-temporal patterns of stimulation [38]. For instance, in medium spiny projection neurons (MSPN) of brain striatum, LTP requires activation of NMDAR by glutamate (Glu) and dopamine receptor D1 (DRD1) by dopamine (DA), termed *dopamine-dependent synaptic plasticity* [39]. Integration of these two signals occurs intracellularly through interacting pathways that modify gene expression, or increase the input signal by insertion of α-amino-3-hydroxy-5-methyl-4-isoxalone propionic acid receptors (AMPARs) into the cellular membrane [40].

1.1.4 Synaptic plasticity in neurobiological diseases

Wide range of neurobiological, cognitive and psychiatric disorders can be linked to aberrations in synaptic transmission and synaptic plasticity [41–43]. Among these are Alzheimer's disease, Parkinson's disease, autism spectrum, Attention Deficit Hyperactivity Disorder (ADHD), schizophrenia, addiction, multiple sclerosis and chronic pain [42]. Many molecular mechanisms involved in manifestation of neurobiological disorders are related to proteins expressed in synapses and processes underlying various types of synaptic plasticity. For instance, two closely associated proteins to Alzheimer disease, tau protein and phosphorylating it glycogen synthase kinase 3 beta (GSK3B), are involved in induction of LTD thereby point to causal relations between abnormalities in LTD and Alzheimer's disease [42, 44]. The same proteins can be linked to multiple disorders. For instance, the protein family voltage-dependent calcium channel, CACNA1C and CACNB2, and two families of dopamine receptors, DRD1 and dopamine receptor D2 (DRD2), are associated with ADHD, autistic disorder, Alzheimer's disease, Parkinson's disease, and schizophrenia [45] (*Appendix D*). Regulation of synaptic proteins by PTM and binding events me-

diating complex composition are known to play central mechanistic roles in healthy synaptic transmission [46]. For instance, aberrations in mammalian target of rapamycin complex 1 (mTORC1) that controls protein synthesis, is related to heritable disorders such as Fragile X or Rett syndrome [46]. Levels of neurotransmitters directly influence the synaptic response and even though not fully understood, their availability is a major target of today's therapeutics [42]. For instance, standard treatment of Parkinson's disease aims to restore levels of DA by administration of L-DOPA, precursor of this neurotransmitter [47].

1.2 System-level approach to study molecular processes

With increase of efficiency in computational techniques and development of hight-throughput technologies that supply large quantities of experimental evidence, an integrative approach to study biological processes appeared as a single but broad interdisciplinary domain of *systems biology*. Contrary to reductionistic approach, system-level approach drives from observations that even simple relations or rules between multiple elements give rise to non-additive properties and complex dynamic behaviour [48]. In molecular signalling, among these complex and emerging properties we can find bistable switch in abundances of molecular species triggered by distinctive output thresholds, positive and negative feedback loops forming self-regulating circuits, and robustness to perturbations [49]. These properties are not easy to infer by studying molecular entities in isolation [48]. To uncover causative mechanisms driving biological processes and disease conditions, two major perspectives in studying systems biology emerged [50]. The first one, data-driven or "bottom-up" approach, originates from rapid development of high-throughput and cost-efficient large-scale experimental technologies, collectively called *omics*-techniques [51]. These experimental technologies are applied to investigate genetic variants associated with complex disease across whole genomes (genomics) [52], examine types and counts of expressed genes (transcriptomics), determine protein identities, their quantities and interactions (proteomics), and identify phosphorylated protein sites (phosphoproteomics) [51, 53]. Dependent on the experimental design, these data sets are collected

from samples of different disease conditions, cell-lineages and under drug-induced perturbations to gain insight into context-dependent alterations in molecular components and biological functions they are involved in. Attempts aiming to analyse integrated multiple omics data sets ("multi-omics") promise more precise identification of functional processes responsible for disease states [51, 54]. These experiments commonly yield differentially expressed lists of genes or gene products. To identify what biological process or molecular pathways are affected in the condition under study, these lists of molecular entities are referred to pre-existing knowledge-bases that organise biological concepts and annotate them with molecular entities based on evidence sourced from research papers. Identified biological concepts divide these molecular entities under study into associated or similar groups with respect to divers biological categories. Among such knowledge-bases are repositories of molecular pathways, ontologies associating genes with molecular functions, biological processes and cell compartments, disease-associated genes, and molecular interactions. Associations between molecular entities are investigated by using formal approaches based on statistical techniques, network graphs, clustering methods and enrichment analysis [55]. These formal procedures assists in identification of core processes, molecular pathways and functions manifested by differentially expressed genes in a disease state [55, 56]. For instance, analysis of high-throughput omics data with network-based associative approach is one of important pillars of personalised medicine [57].

Other important perspective in systems biology is mechanisms-driven or "top-down" approach of dynamic models [50]. Construction of dynamic models can accommodate typically a fraction of molecular components analysed with data-driven approaches. Nevertheless, dynamic or kinetic modelling allow to study biological phenomenons with mechanistic and quantitative details [58]. Specification of dynamic models requires such details as a list of biochemical reactions between molecules that regulate their concentrations and kinetic parameters defining speed of these reactions. Formulated with a mathematical formalism such as a system of ordinary differential equations (ODEs), kinetic models can replicate dynamic behaviour of interacting molecules recorded as quantitative variations of molecular concentrations over time, *time courses* or *trajectories*.

Precise knowledge of mechanisms is particularly important in devel-

opment of new therapeutics. In spite of massive investments and extensive exploitation of omics technologies, drug discovery is facing a high attrition rate where a large number of clinical project fail to result with an approval of new therapeutics. Longitudinal studies at AstraZeneca have shown that lack of efficacy was a major reason of project closures in later phases of drug development [59]. Though the advent of high-throughput screening increased the number of new targets, the number of newly approved drugs remained the same. Furthermore, data obtained with high-throughput drug-screenings remain largely unexplored [57]. The lack of efficacy can be attributed to an incomplete understanding of what are drugs effects on multiple levels that lead to various unpredicted and often unwanted clinical outcomes such as drug resistance and side effects [57]. For example, on and off-targets lead to cascading effect of response that propagates horizontally in gene regulatory, signal transduction and metabolic networks, and across different levels of biological organisation, through cell, tissue, organ interactions up to organismal level. To predict consequences of therapeutic intervention induced on the molecular level, combination of multi-omics and network-based techniques with large-scale multilevel modelling have been recognised as a difficult but necessary step in modern pharmacology [57].

Next sections present an outline of formal methods classified to the two approaches in systems biology that are used in this thesis. Different knowledge-bases and reference data sets, crucial in the data-driven approaches in systems biology, are briefly outlined. Their more detailed description is provided in chapter-specific introductory sections.

1.3 Approaches to analyse gene and protein lists

Two major techniques, classified to the qualitative "bottom-up" approaches and applied in this thesis, are network and enrichment analysis. This section presents a general overview of both techniques as irreplaceable steps in identification of biological context underlying mechanisms of diseases.

1.3.1 Network analysis

Network analysis is a part of mathematical domain of graph theory. A network is a structure composed of abstract objects, called nodes or vertices that can represent any category of interest. Associations existing between

nodes are defined as edges or links drawn between network nodes. As such, network analysis is a systemic approach that allows to shift attention from properties of individual entities, like proteins or genes, to relations between large numbers of them [56]. In the field of systems biology, network analysis is a "bottom-up" approach [50] that exposes patterns in large scale omics data sets by connecting molecular entities by experimentally determined associations, such as gene expression or protein interactions [60, 61]. Depending on the study subject, nodes typically represent biomolecular entities, i.e. ligands, signalling proteins, disease genes, enzymes and metabolites. Consequently, edges can denote various relationships between these node types, such as "interact", "co-expressed" or "is associated to a disease". A network can be composed of more than one type of nodes, termed as *bipartite* or *multipartite* graphs for two and mulitple node types, respectively [62]. There might be more than one relation denoted with an edge between two nodes that can be characterised by numerical or categorical attributes. These attributes provide additional level of information that enriches representation and analysis of networks. For instance, based on node or edge attributes, a network can be conditionally filtered to obtain different perspectives of the network components. These attributes can carry experimentally-derived information (e.g. level of gene expression), or properties arising from network-structure (e.g. number of edges connected to a node). Commonly used attributes characterising edges are weights. Networks containing weighted edges are termed *weighted networks*. Weights can be represented with a numerical value to denote importance of relations between nodes [63, p.34]. Network edges can be undirected or directed. In directed networks, edges between node pairs have associated directionality, visualised with arrows. In these kind of networks, nodes are divided into source and target nodes.

Numerous properties and statistics can be derived from the network structure. These statistics summarise either local or global network properties, determined by node or edge-related measures, or characterise general tendencies of the entire network such as topology. Among popular concepts characterising networks and their components are centrality, modularity, connectivity, and distance measures [64]. Based on these concepts, more general network characteristics are defined, such as different types of generative graph models (scale-free, random), topology-based network types (linear, tree, star),

and methods for partitioning network nodes into smaller subgroups, modules or clusters. Definitions of all these terms are often based on most fundamental graph concepts such as a *path* or *node degree*. A path between two nodes is a distance measure defined as a sequence of joined nodes and edges, thereby a *shortest path* is a minimal number of such links. A node degree is a number of edges connected to a node (in- and out-degree in directed graphs). Among many definitions of centralities, the node degree is the simplest concept of network-based centrality measure. In general, centrality measures characterise nodes, edges or subgraphs with respect to differently understood importance measures. A node with large number of neighbouring edges, termed *hubs*, are known to be in some respect more important than sparsely connected nodes. For instance, in networks representing binary interactions between proteins of Yeast *Saccharomyces cerevisiae* interactome, proteins identified as hubs were found to be essential and evolutionarily conserved [65]. Deletion of hub proteins in the same model organism was more likely to produce a lethal effect [66]. Other study found that perturbations of hub proteins caused larger diversity in disease phenotypes [61].

Next to identification of single densely connected hubs, tightly connected groups of nodes are important and biologically meaningful components. By using degree statistics, a network can be divided into densely and sparsely connected regions, *cores* and *peripheries*, respectively [67]. Localisation of disease-associated genes in either of these network regions is correlated with different disease classes [68]. For instance, it was shown that a large number of non-lethal disease genes are found in network peripheries [64, 69].

Among definitions and approaches used to expose densely connected network regions, or cohesive subgraphs, are *cliques, n-cliques, n-clans, k-plexes* and *k-cores* [70]. Among them, cliques are most topologically rigorous groups defined as subgraphs of completely connected nodes [71]. Such strong topographic relationships between nodes usually denote protein-complexes and functional modules [72]. Due to insufficient knowledge of protein interactions, clique identification might be affected by a high rate of false-negatives and false-positives [71]. Therefore, different relaxation methods of the clique concept were introduced, such as: γ-quasi-cliques, n-cliques, n-clans and k-plexes. Compared to others, *k-cores* are most relaxed approach to identify densely connected subnetworks [70]. Algorithmic implementation of k-cores identification

is more efficient than in other approaches [73]. The k-core of a graph is obtained by gradually removing vertices of degree $< k$. This procedure results with a subgraph, or disconnected subgraphs, that have nodes of degree $\geq k$. The main-core is a subgraph with a maximal value of k-parameter [73]. By removing least connected nodes, the central cores of densely connected graph components are exposed. Stratification of networks by choosing a gradually higher k-parameter was used to efficiently visualise large-scale networks [74].

The above concepts identify maximal cohesive subgraphs with respect to specified property in the network graph. Though a cohesive subgraph is a form of community, another class of methods to efficiently partition all network nodes to communities were introduced in graph-based analyses [75]. Nodes associated to a community form subgraphs that have tighter connections within their community than to other communities [76]. Partitioning of molecular networks into communities serve as proxies for identification of functionally related proteins, diseases or physical modules [61, 64]. There is no universal definition or a finite set of principles by which to define what is a community or cluster and therefore, there are many clustering methods [77]. In a recent comprehensive survey of community detection methods by Javed et al. [78] clustering methods are classified into *disjoint communities* and *overlapping communities*. In the first case, a node is exclusively associated to a single cluster, whereas in the other one, a node can be a member of more than one cluster. Among methods identifying disjoint communities, the reviewers list three main categories of algorithms: traditional, modularity-based and dynamic. Traditional algorithms initiated the very first concepts and methods of clustering that provided foundations for latter introduced categories of algorithms. An important class of traditional community detection method is *hierarchical clustering*. It can be divided into top-down (divisive) or bottom-up (agglomerative) approaches, dependent on node-to-cluster assignment in the initial step of iterative procedure. In the divisive class of algorithms, all nodes are assigned to a single cluster that is gradually partitioned into smaller subgroups based on a similarity measure between nodes. In the agglomerative algorithms, each network node is a singleton cluster that is merged with its neighbouring cluster-nodes based on a predefined similarity, distance measure or a strategy. An important example of the agglomerative approach combined with a strategy is the Girvan–Newman algorithm that is based on removal of

edges that are most "in-between" communities [76]. The computational cost of the Girvan-Newman algorithm, impractical for large networks above 1000 nodes [79], led to proposition of an alternative method based on maximisation of modularity score by Newman [80]. This became an important class of community detection algorithms [78]. Modularity is a global network property that measures cluster quality [81]. It is defined as a number of edges within groups minus a number of expected randomly distributed edges [75]. Optimisation of a cluster modularity score is computationally hard [82] and therefore, approximation methods that use local (greedy) optimisation techniques were developed. These algorithms are based on the hierarchical agglomerative approach that at every iteration merge two separate clusters if these increase the modularity score [81]. The algorithm proposed by Blondel et al. [82] is considered as most efficient and scalable up to date [78].

Efficiency of an algorithm is an important aspect, being one of two eminent criteria in algorithm comparison, the other one being accuracy. Systematic comparison of clustering methods is a difficult task. It commonly relays on benchmark generative networks combined with information recovery metrics, e.g. Adjusted Rand Index (ARI) [78, 81]; or real-world networks of unknown ground-truth clustering evaluated with cluster quality measures, e.g. modularity [81]. The study by Emmons et al. [81] found that different information recovery metrics and cluster quality measures disagree on performance evaluation of the same set of algorithms. Moreover, the authors admitted that high performance obtained by the algorithm that they identified as the best one, tested on a generative benchmark model, might be due to similarity of a model underlying the algorithm and the benchmark network [81]. On the other hand, testing accuracy of algorithms on real-world networks with cluster quality measures does not offer any direct answer on correctness of clustering results. These type of networks might not even have uniquely defined communities, and evaluation of communities requires support of metadata information on cluster members to resolve quality of clusters [79]. It is commonly admitted that with the variety of network structures, sizes, and metrics for clustering evaluation, categorical statement announcing superiority of any particular clustering method over others is yet to be achieved [79, 81].

1.3.2 Enrichment analysis

High-throughput experiments, such as performed with DNA microarrays, measure levels of messenger RNA (mRNA) transcripts as manifestation of expressed genes in samples originating from different tissues, cell types and disease conditions [83, 84]. These experiments yield differentially expressed lists of genes in the sample of interest that are either over, under or neutrally expressed compared to the reference sample [85]. To learn what biological functions or processes these lists of genes are involved in, *enrichment analysis* is performed against categories of a reference database. Associations to biological concepts are also desired to be identified amongst genes associated to a disease of interest, collected from various published studies in the subject, such as genome-wide association studies, candidate-gene association studies, linkage studies, mutational studies, genome-wide copy number variation analyses and meta-analyses [86].

Based on evidence, each category in a reference database is associated with a list of genes, where a gene can be a member of more than one category. This thesis applies over-representation analysis (ORA), the simplest and most commonly used approach for enrichment analysis [55]. ORA provides evaluation of statistical significance that asses if proportion of differentially expressed genes, associated to predefined biological categories, is larger than expected by chance. To evaluate this significance, the hypergeometric distribution or the one-tailed Fisher's exact test is commonly applied, both known to be equivalent measures [87].

ORA has well-known limitations. Unlike Functional Class Scoring (FCS) [55], ORA takes into account only the number of genes that was significantly expressed above a predefined threshold value. Differential gene expression measured with microarrays assigns values to genes that denote change in their expression. This information is not considered in identification of enriched categories with ORA. For the purpose of this study, this is a more suitable property as genes subjected to enrichment analysis are not associated with any weighting values.

Enrichment analysis is primarily possible owing to predefined annotations collected in reference databases. These databases link genes to various categories that are often organised into *ontologies*. Ontology as a structured and machine-readable representation of interlinked biological concepts that

are based on controlled and uniform vocabularies specific to the subject area [88]. Sharing ontologies improve interoperability and integration across resources [88, 89]. One of the most prominent catalogue of controlled vocabularies and categories annotating gene products used for enrichment analysis is Gene Ontology (GO) [90, 91]. GO is abundantly cross-referenced and links genes to three types of categories: molecular function, biological process and cell compartment. Each category is composed of hierarchically organised biological terms linked with each other by multiple kinds of relations (e.g. "is a", "part of", "regulates"). Each term is annotated with genes based on evidence derived from biomedical publications [91].

Other commonly used reference dataset are databases annotating gene products to molecular pathways [55]. Pathway maps or diagrams are important sources of detailed information on reactions between molecular entities. Molecular pathways are divided into three groups: metabolic, signalling and gene regulatory [92]. Kyoto Encyclopedia of Genes and Genomes (KEGG) [93] was one of the pioneering initiatives that issued a web-based and publicly available expert-curated diagrams of different molecular pathways, observed in multiple taxonomic species [92]. Since then, numerous commercial and academic resources of molecular pathways have appeared, often specialising in one of the pathway types. A review by Chowdhury and Sarkar [92] analyses a selection of 24 open-access pathway databases. The reviewers divided pathway resources into self-curated primary databases, secondary databases aggregating multiple resources, and hybrid databases that are combinations of the first two. In the first category, arguably most commonly used databases are REACTOME Pathway Database (REACTOME) [94] and KEGG. Examples of the second and third categories are PathwayCommons [95] and Wikipathways [96], respectively. For extended view, pathway databases enrich pathway maps with other integrated resources, such as protein interactions, phosphoproteomics or gene expression [92]. Unification of pathway resources is an outstanding challenge [92] despite existence of standard community pathway formats as Systems Biology Markup Language (SBML) [97] and Biological Pathway Exchange (BioPAX) [98]. Multiple individual studies attempted pathway data consolidations on different levels [99–102]. Nevertheless, these initiative are often discontinued and result in partially integrated resources. Among major obstacles of such consolidation are incompatibile data models, differ-

ences in nomenclature, pathway names, molecular species and interaction sets that define a certain pathway, non-uniformly accepted and applied standards in pathway couration and used ontology [92, 100]. Even agreement on using a uniform pathway ontology across resources would be a step towards a global pathway network [92, 103]. Integration of pathway datasets has to be an effort of the whole community as this requires general agreement, trust and recognition of introduced standards.

1.4 Dynamic modelling of signalling systems

Dynamic modelling is the second approach defining systems biology. It aims to study mechanistically and quantitatively complex behaviours of interacting molecules. This cannot be achieved by analysis of omics datasets with network or enrichment analyses, as both methods identify links to biological mechanisms and processes rather than explore them in detail. Dependent on the availability of details regarding the biological system under study, dynamic modelling give means to computationally explore how perturbations can affect concentrations of molecules and alter behaviour of the system. This approach could contribute to development of mechanism-based therapies with more predictive power on outcomes of pharmaceutical interventions [57].

Dynamic computational modelling frameworks consist of a model specification and simulation methods. Model specification is a set of equations or instructions written in a machine-interpretable language. These languages are based on mathematical formalisms and define relationships between variables that change in quantities is measured over time. This is achieved with a numerical simulation, defined as a computational procedure that realises the principles encoded in the model over specified limit of time.

With suitable abstraction and expressible language for system description, experimentally-derived evidence can be encoded in formal machine-readable representation and subjected to an executable procedure that outputs realisations of system's temporal dynamics, comparable to experimental observations. Suitable formalism should encode elementary facts in unambiguous, and explicit notation thereby they remain disputable. The formalism of choice should allow for a direct rather then interpreted description of biological phenomenons. Because the model should flexibly accommodate more than one hypothesis, its notation should be easy to modify. Moreover, the concept and

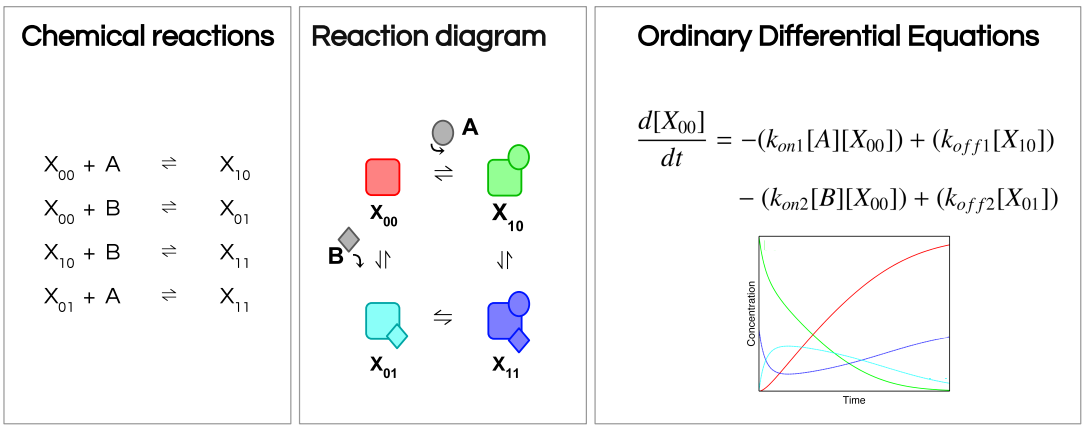


FIGURE 1.1: The first step in building a kinetic model of molecular interactions is definition of *chemical reactions* between molecules. These reactions can describe chemical transitions between states of one reactant catalysed by the other. An arrow denote direction of the transformation and a double arrow, reversibility of a reaction. A set of chemical reactions can be depicted with *reaction diagram* that are converted to their mathematical representation of a coupled (simultaneously solved) system of ODEs. Each rate equation expresses a change of one molecular species concentration over time, formulated with reaction rates that directly influence the species change [104]. In this figure, a rate equation for X_{00} is defined with four rate laws of mass action for each influencing reactions. The law of mass action states that the speed of reaction is proportional to the product of the concentrations of the reactants [105, p.141]. Positive and negative signs of reaction rates denote direction of arrows pointing respectively towards and away of X_{00} [104]. Example adopted from Hlavacek et al. [106].

notation ought to be intuitive and transparent to ease its understanding. It is especially important in the domain where knowledge spans across disciplines. Dynamic models can integrate variable data sources that have yet to gain place in the body of knowledge [48], thereby enabling consolidation of knowledge and falsification of assumptions.

In this section a classic equation-based approach to dynamic modelling of molecular systems is presented, followed by introduction to alternative methods of modelling originating in computer science.

1.4.1 Ordinary differential equations

Dynamic modelling of molecular pathways has been traditionally achieved by solving a set of coupled ODEs. The ODE-based modelling is acclaimed formalism with a long tradition [104]. Contrary to partial differential equations

(PDEs), ODEs are equations formulated for a single independent variable. In the models in systems biology this variable commonly denotes time. Equations for a kinetic model are defined according to a set of chemical reactions that describe how reactants turn into products [107] (FIGURE 1.1). These chemical reactions are often drawn as *reaction diagrams* to conceptually represent transition rules in the system. Growing number of molecular reactions that compose modelled pathways results with a larger number of elements that have to be represented in such diagrams. This yields difficulties in reasoning and understanding of behaviour of molecular reaction systems [108]. Such diagrams are rather an auxiliary and intermediate step towards quantitative evaluation of model behaviour achieved by converting these coupled chemical reactions to a set of ODEs. The number of rate equations is equal to the number of reacting species. Each rate equation expresses the change of concentration of a single molecular species over time. A rate equation is formulated with rates of reactions that directly take part in creation and elimination of this species [104]. An example of a rate equation in FIGURE 1.1 is based on rate laws of mass action. Each reaction rate is weighted by a *rate constant*, specific to each reaction and dependent on temperature.

As real-size molecular systems are not analytically solvable with a pen and paper, simulation of ODE-based models requires computer-aided calculations [48]. Hence, similarly to Knüpfer et al. [109], ODE-based models are understood here as a type of computational models. ODE-based models are routinely solved with numerical procedures, e.g. Runge-Kutta method [107]. A strong point of modelling with ODEs is a large number of mathematical methods for model analysis, among which we can find stability analysis, fixed points, phase planes, nullclines, rate balance plot, and bifurcation analysis [110]. Moreover, extensive standard and software development support and facilitate formulation and analysis of ODE-based models [104, 111, 112].

Time courses obtained by solving ODEs models are deterministic and continuous, characterised by smooth and gradual change of species concentrations over time [113]. Calculation outcomes are unchanged for the ODE model with the same initial molecular states and input parameters. This setup does not reflect the actual characteristics of subcellular events driven by random collisions between discrete molecules [114]. Despite being an approximation of the real molecular process, the ODEs-based modelling method has been

successfully applied and founded the baseline approach to dynamic modelling in systems biology. However, the deterministic and continuous approach is correct as long as abundances of reactants are large enough to render random fluctuations as negligible [115]. There are molecular properties and mechanisms that cannot be modelled with this approach, such as transcriptional bursting [116] or DNA damage [113]. Molecular processes in synapses are characterised by high numbers of divergent types of molecular species but low numbers of instances per each type. In these conditions, magnitudes of fluctuations in copy numbers of molecules becomes important [115]. Existence of noise and its role in shaping characteristic properties of signalling systems are acclaimed and closely studied [117]. Among others, inherent presence of noise in signalling systems is attributed to transient, low-affinity and promiscuous protein interactions [118]. These variations in species counts can only be captured with the stochastic simulation [115]. However, rate equations can be expressed stochastically with stochastic differential equations (SDEs) [113]. Moreover, current development in standard formats encoding biological models, in particular Systems Biology Markup Language (SBML), allows to obtain trajectories of the same model with different simulation methods, both deterministic solvers and stochastic simulators, e.g. Gillepie's Stochastic Simulation Algorithm (SSA) [112, 119]. More important weaknesses of ODE-based models lays in requirement of explicit enumeration of molecular species for which rate equations are defined [120]. This precludes modelling of molecules characterised by combinatorial explosion of possible states and numerous binding partners, observed in such examples as EGFR (*Section 1.1*). EGFR is an important family of protein receptors, targeted in cancer therapies [121]. When activated, the receptor is autophosphorylated at multiple Tyrosine sites and dimerises to yield a large number of possible states and functionally different conformations [18]. EGFR binds and activates different protein targets propagating extracellular signal through intracellular pathways. EGFR is one of the most studied examples that remains intractable due to variability of consequences of its multiplicity of states [122, 123]. Equations can be written for only a fraction of all molecular species of EGFR [124].

1.4.2 Methods inspired by computer science

Development of formal methods in computer science has prompted attempts to use them in modelling molecular processes. These approaches extended the number of observed properties of biological systems that can be dynamically and quantitatively modelled [125]. These approaches are broadly reviewed by Bartocci and Lió [125], Machado et al. [126], Tenazinha and Vinga [127] and Ji et al. [128]. Computational methods are commonly divided into temporal, where well-stirred medium of molecular events is assumed, and spatio-temporal, that model impact of spacial distributions of molecules [125]. Some of these methods were more commonly used to model particular types of pathways, such as signalling, gene regulatory or metabolic [126]. These methods can reflect different aspect of the same molecular processes by addressing a subset of characteristics found in multiple biological system. There are also methods that support versatile modelling features and tools. Methods are also divided with respect to the level of provided detail, ranging from modelling qualitative state transitions (e.g. Boolean networks) to detailed mechanisms (e.g. process algebras) [129]. For the purpose of this study, non-spacial mechanistic modelling methods are considered that allow to define processes on subcellular-level, as oppose to multiscale models. The multiscale modelling is a very challenging task. The major concern in this type of modelling is creation of an environment that integrates numerous baseline low-level formalisms. A general difficulty lies in designing synchronisation strategies to integrate output responses and time-scales of multiple connected modules defined with low-level formalisms. These can either model different levels of detail [130] or combine different levels of response, ranging from molecular, cell to tissue. In systems neurobiology, there are examples of combined use of NEURON and ECell [131], and NEURON with Kappa [132]. Next to these experimental studies, there are also more established frameworks such as MOOSE (Multiscale Object-Oriented Simulation Environment), a general-purpose biological simulator that at the ground level is based on ODEs.

This study is concerned with modelling methods that provide a low-level definition of molecular interactions. Such formalisms commonly evolve to encompass more complex problems that concern space [133], molecular geometries [134] and multiscale processes [132].

Among most popular methods that fit these characteristics and offer

most versatile collection of features are *Petri nets*, *process algebras* and *rule-based* formalisms [126].

Petri nets are one of the first methods developed to study parallel and distributed processes in computer science [135]. Among advantages of Petri nets are intuitive graphical representation and methods for model analysis that are based on matrix algebra. A Petri net model is represented as a bipartite graph with two types of nodes, *transitions* and *places*. In a bipartite graph, only nodes of different types are connected. Place nodes denote reactants whereas transition nodes, reactions. Reaction execution triggers token passing that can be translated to concentrations or copy numbers of interacting molecules. There are multiple variants of Petri nets, both for qualitative and quantitative (continuous, stochastic) modelling. However, Petri nets are unable to model multi-state molecules characterised by combinatorial explosion of molecular species as all network nodes (reagents) have to be enumerated beforehand. Moreover, the advantage of graphical representation loses its strength when the modelled system becomes large.

Following the invention of Petri nets, different *process algebras* were developed as formal languages to model and analyse distributed, interacting processes in computer science [135]. On this canvas, large family of languages for modelling biological systems emerged [136]. Differences between these languages lie in the choice of what constitutes a basic element of biological process [136]. For instance, π-calculus [137], Beta-binders [138], and BlenX [139] (derived from Beta-binders) present a *molecule-centred view*, whereas BioPEPA [136], a molecular species or *reagent-centred view*. Reagent-centred languages such as BioPEPA, amongst other goals, aimed to match representation of biochemical networks defined by the SBML standard format for biological models (Level 2) [140]. Being compatible with the SBML format, it was not designed to tackle combinatorially complex assemblies of molecular species. Only recently a plugin to the SBML format, *multi*, was introduced that contains data structures to represent molecular entities with multiple states and components [141]. Only languages with molecule-centred view were able to model such processes as self-assembly of actin polymerisation [142, 143]. Actin polymers are grown by complexation of monomeric subunit of actin proteins. In reagent-centred process algebras, gradually assembled polymers would have to be represented as distinct molecular species.

What is in common for the whole family of process algebras is that it offers expressible formalism to represent, model and analyse biological processes in modular and compositional way by describing modelled systems with their components [144, 145]. Among a range of general biological aspects that can be symbolically represented and modelled with process algebras are cell compartments [146], protein domains [137] and polymerisation [142, 143]. However expressible process algebras can be, since they were originally developed in the other than biological context, concepts embedded in some variants of these languages might be non-intuitive and obscure to comprehend for biologists [145] or might simply be redundant [147]. This is true even for adjusted process algebras for the purpose of molecular modelling [136, 139]. To such examples belong a concept of *channels* in π-calculus, that are shared between processes (molecules) in a pair-wise communication [137].

Another approach to model biological systems is rule-based (RB) modelling. It belongs to a more general class of agent-based modelling [148] that was first brought about to study problems outside of the systems biology context [149]. In the rule-based setting, actions of individual agents are defined by a set of local and partially complete rules. In this way, a large set of system behaviours can be represented by a much smaller number of general rules that is the most prominent feature of RB modelling [149]. Among others [150], arguably most popular representatives of RB modelling methods in systems biology are based on *graph rewriting*: Kappa [147] and BioNetGen [151]. Both form well-equipped frameworks that popularly exemplify RB modelling in reviews presenting modelling formalisms in systems biology [126, 152–154]. Moreover, they offer an intuitive interpretation [155], where graph transformations formalise behaviour of agents represented as graphs. Graph rewriting was developed to support operations on theoretical structures that are more general than strings [156]. In the realm of computer science, a graph rewriting can be understood as operations on graph data structures that evolve according to a set of local instructions [156]. In the context of molecular modelling, methods based on graph-rewriting represent a molecule-centred view, where molecules are structured graph objects, i.e. nodes with sites. Reactions occurring between molecules are represented as graph-transformations, where bonds can be formed between sites of nodes. Graph transformations are encoded as rules that are instructions for local transformations. In this way a

rule can represent a set of reactions or an exact reaction instance, dependent on the rule specificity. This constitutes a significant advantage of rule notation that can express an infinite number of reactions with a small and finite number of rules. In the most of modelling methods mentioned above, every chemical species has to be specified in advance what is highly problematic for species with dozens of phosphorylation sites and many possible states. This limiting factor makes these methods inappropriate for modelling complex interactions of signalling proteins.

The ability to capture a protein as a graph with binding sites (e.g. domains) that have internal states (e.g. phosphorylated) gives a sufficiently expressive system to capture principal mechanisms of signalling processes (e.g. dissociation, synthesis, degradation, binding and complex formation [152]) as well as insight into site-specific details of molecular interactions: affinities, dynamics of post-translational modifications, domain availability, competitive binding, causality, and the intrinsic structure of interactions.

A review of other rule-based modelling methods used in systems biology that rely on other than graph-rewriting formalisms can be found in Stefan et al. [150].

1.5 Rule-based modelling

In this project, RB modelling was found as a most flexible and suitable formalism, tailored for stochastic modelling of site-specific protein interaction networks. As an example of the RB modelling approach, the Kappa framework is applied in this thesis. As a well-designed modelling framework, the specification language is separated from the simulator. The framework consists of the Kappa language and the KaSim simulator. Each of the two components is separately described in this section.

1.5.1 Kappa language

The Kappa language formulates principles of the model that combines process algebras and graph transformations to create a tailored formalism for protein interaction networks [147]. The next paragraphs will informally present definitions of syntax and concepts necessary to understand result chapters of this thesis. The syntax of rules described here is relevant to the 3.5 version of the KaSim simulator [157].

1.5.1.1 Agents

An *agent* is a fundamental object in rule-based models that represent molecular entity, e.g. a protein (Figure 1.2A). Agents are atomic model components that preserve identity [120] and cannot be decomposed any further [158]. An agent is defined by a name and *interface*, together forming an *agent signature* (Code 1.1). The interface is a finite set of named *sites*. Each site is assigned with a unique *label* that can simultaneously carry 2 types of information: *internal state* and *binding state*. Each site can have multiple internal states and binding partners during the simulation but only one of each at a time. Internal states per site, by convention, are denoted with character letters following the site label and the "~" sign. Code 1.1 shows an example of the agent signature. "A" is an agent name. Its interface is enclosed in brackets. The interface consists of one site labelled as"pSite" that can have two internal states: "~u" and "~p". By convention, the binding state is denoted by an integer following the site label and the "!" sign. There are 4 possible binding states that the site can be in: free, semi-linked (bound to an unknown site), unspecified (free or bound) or with a labelled bond (bound to a named site) (Code 1.2). A bond label can exist exactly twice, shared between two sites. This implies that only binary bonds are allowed [158]. The order of writing sites and agents is of no consequence. Edge labels do not have to be identical within the model as long as their uniqueness within the rule expression is satisfied [158].

Code 1.1: Example of agent signature specification

```
%agent: A(pSite~p~u)
```

Code 1.2: Agent's possible binding states in a state p

```
A(pSite~p)          # free
A(pSite~p?)         # unspecified
A(pSite~p!_)        # semi-linked
A(pSite~p!0)        # bound with a labelled bond
```

1.5.1.2 Patterns

Two or more bound agents form a *complex*. The complex belongs to a more general category of *expression*. The expression is composed of comma separated agents and complexes [159]. All agents and complexes at a given point

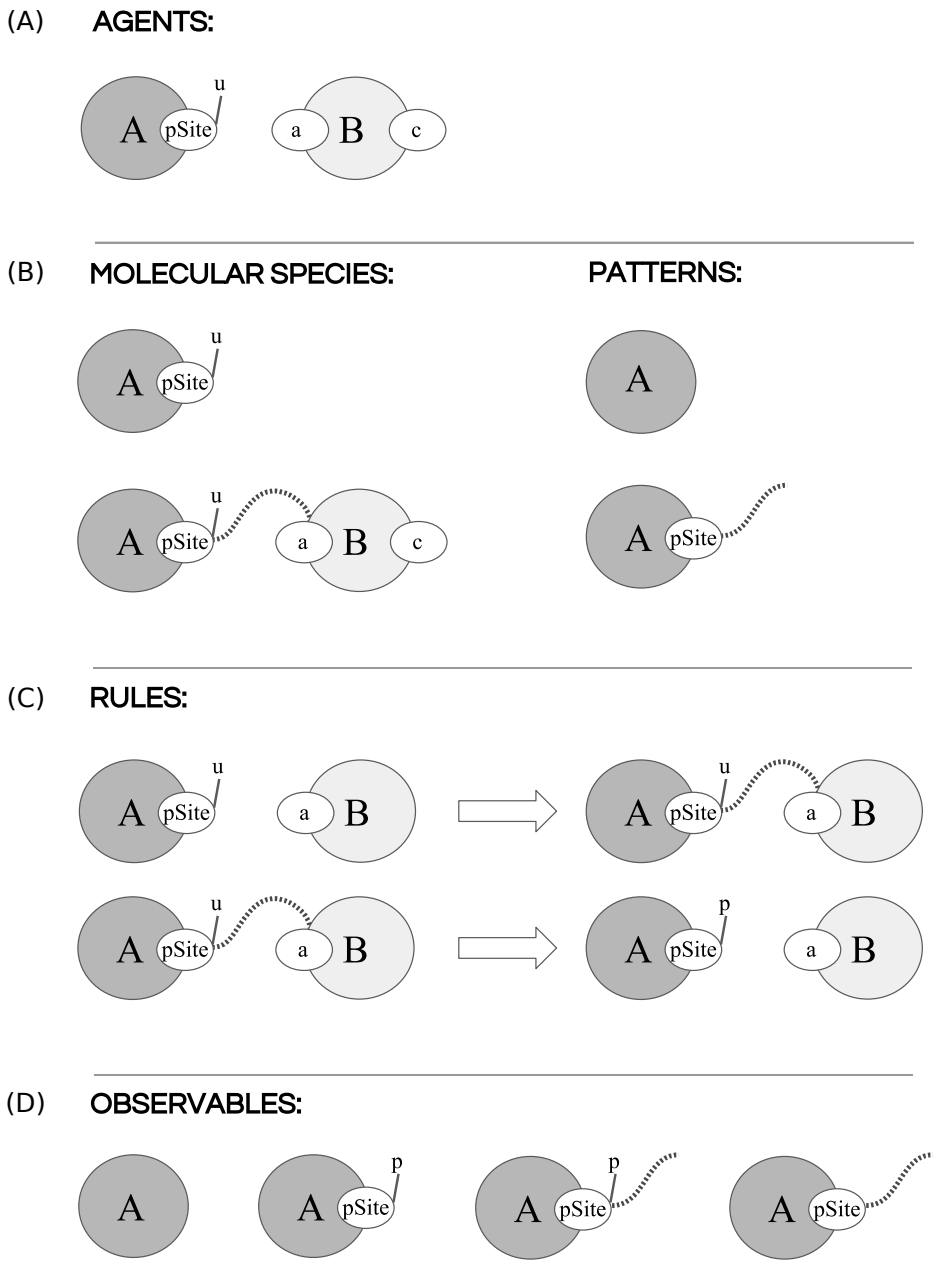


FIGURE 1.2: Diagram demonstrating main concepts and elements of the Kappa language. (A) *Agents* represent molecular entities as named nodes (*"A", "B"*) with sites (*"pSite", "a", "c"*) and states (*"u"*). (B) *Molecular species* are agents and complexes that contain full description of their states and site occupancy. As such, they can only denote their one particular type. *Patterns* offer partial description of agents and complexes thereby match larger groups of molecular species. (C) *Rules* define transformations of agents understood here as reactions. Dependent on the generality of agent description, a single rule can represent a set of reactions. (D) *Observables* are desired simulation outputs that can be represented as patterns or molecular species.

of simulation form *mixture* of existing *molecular species* in a system. A molecular species is an agent or complex made of completely specified binding and internal states of agents that exist in the mixture. Agents can have completely or partially specified interfaces. Agents with partially specified interfaces are termed expression *patterns* [159]. The pattern can be less or equally specific to completely determined expressions of molecular species (FIGURE 1.2B). In the former case, the pattern allows to omit information that is irrelevant or unknown regarding agents states [159]. Compare CODE 1.3 and CODE 1.4.

CODE 1.3: Agents with partially specified interfaces (patterns)

```
A()              # unspecified internal and binding states
A(pSite?)        # unspecified internal and binding states
                 # (equal to the above)
A(pSite!_)       # unspecified internal state and binding partner
A(pSite~p?)      # internal state 'p' and unspecified binding state
```

CODE 1.4: Agents with completely specified interfaces (molecular species)

```
A(pSite~p)       # in state 'p' and unbound
A(pSite~u)       # in state 'u' and unbound
A(pSite~p!0)     # in state 'p' and bound with a labelled bond
```

1.5.1.3 Rules

Two expressions separated with an arrow, e.g. "<-", "->" or "<->", form a *rule*, composed of a left-hand site (LHS) and a a right-hand site (RHS) [158] (FIGURE 1.2C). A rule is an instruction that expresses an action or state transformation [156]. The type of an arrow determines direction and reversibility of the transformation. Expressions forming a rule can be defined with patterns. Less detailed rule pattern matches to more instances in the mixture of molecular species thereby can represent more then one reaction. This pattern matching can be formally understood both as string matching or graph *embedding*. As the latter, it should satisfy injections on graphs with preservation of elements of agent interfaces, i.e. names, sites, internal states and bonds [120]. Pattern matching in graphs is embedding of a less detailed graph in the rule specification to a completely specified graph of a rule instance. Omission of details regarding agent interfaces in the rule specification indicate that these details does not restrain reactions to take place. The transformation instructed

by a rule is performed by matching expression on LHS to instances existing in a mixture of molecular species. An instance that matches a rule is transformed according to the expression on RHS. Given that agents and complexes are understood as graphs, rule application is a graph rewriting procedure. Elementary level transformations of molecular graphs expressed with rules are: synthesis, degradation, binding, dissociation and state change.

A rule specified with complete information about agent interfaces is in one-to-one correspondence to a chemical reaction. It is illustrated with the following set of chemical reactions of two-step dephosphorylation of the agent "A" by the agent "B":

$$\mathrm{Ap} + \mathrm{B} \longrightarrow \mathrm{Ap}{-}\mathrm{B} \tag{1.1}$$

$$\mathrm{Ap}{-}\mathrm{B} \longrightarrow \mathrm{Ap} + \mathrm{B} \tag{1.2}$$

$$\mathrm{Ap}{-}\mathrm{B} \longrightarrow \mathrm{A} + \mathrm{B} \tag{1.3}$$

$$\tag{1.4}$$

Equivalent rules to the above chemical reactions being in one-to-one correspondence are as follows:

```
1 A(pSite~p  ),B(a  ) -> A(pSite~p!1),B(a!1)
2 A(pSite~p!1),B(a!1) -> A(pSite~p  ),B(a  )
3 A(pSite~p!1),B(a!1) -> A(pSite~u  ),B(a  )
```

1.5.1.4 Observables

In the RB modelling, outputs of interest resulting from the model simulation and represented as time courses are defined in the model as a list of *observables*. Similarly to the rule definition, an observable can be represented with different levels of detail (Figure 1.2D). The level of detail can range from generic patterns that supply partial information on agent interfaces, to their complete description that matches the definition of molecular species. An observable declared as incomplete pattern refers to multiple molecular species that time courses are summed to represent a single trajectory denoted by this observable of interest. For instance, Code 1.5 exemplifies an observable that represents a sum of trajectories of the agent "A" that are phosphorylated and in an unknown binding state.

Code 1.5: Example of observable definition

```
%obs: 'Ap' A(pSite~p?)
```

1.5.1.5 Perturbations

The Kappa language provides means to induce perturbations during the simulation. This extends control over the simulation and allows flexible design of experiments. Perturbations can be applied once or repeatedly when or while indicated preconditions are satisfied. A template of a command for a one-time perturbation is shown in *Code 1.6*.

Code 1.6: Command template for one-time perturbation

```
%mod: <boolean expression> do <effect>
```

The `%mod:` operator initiates the perturbation command. `<Boolean expression>` is a part of the command where preconditions are located. If the expression is evaluated as true, the `<effect>` is executed. The boolean expression may refer to a specific moment of the simulation, counted in seconds (`[T]`) or simulation *events* (`[E]`). An event is defined as a rule application that transforms molecular species in the mixture in progression of the simulation [157]. The time when perturbation is induced can also be formulated as a moment when an agent's or a ratio of agents' copy numbers reaches some predefined level. A command template for repeated perturbation is shown in *Code 1.7*.

Code 1.7: Command template of repeated perturbation

```
%mod: repeat <boolean expression>
      do     <effect>
      until  <boolean expression>
```

In repeated perturbations, `<effect>` is executed until the second `<boolean expression>` is evaluated to false. A complete list of effects can be found in the Kappa manual [157]. Effect can be used either to apply changes to the model or explore the simulation by collecting additional data during the model execution. In this study, two effects were used to update rate constants and to add molecules. The former with the `$UPDATE` command, the latter with the `$ADD` command. Both result in alteration of molecular abundances during the simulation. To collect additional data during the simulation, *snapshots* were used to capture the state of molecular mixture at an indicated time point and

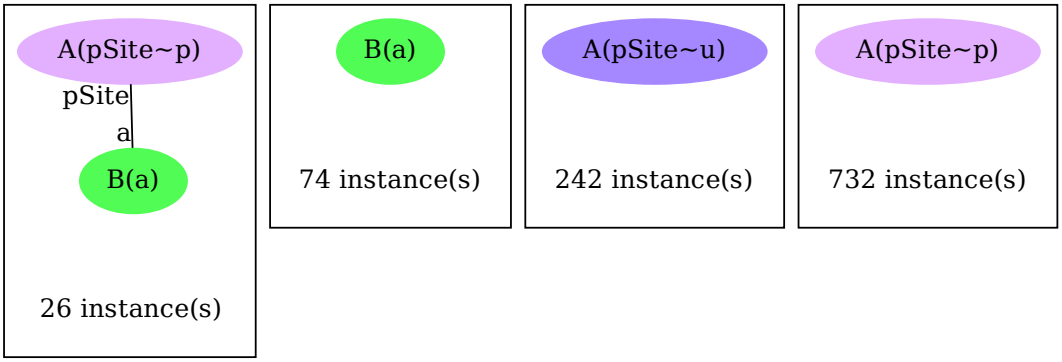


FIGURE 1.3: A snapshot generated with the Kappa framework that represents the molecular mixture of the RB model example discussed in this section. Each rectangle contains molecular species represented as connected graphs. Agents are represented with nodes of different colours and bonds between agents with edges. Copies of the same agent composing molecular species are represented with the same colour. Number of instances per molecular species are indicated at the bottom of the species rectangle.

executed with the `$SNAPSHOT` command. Snapshots provide means of monitoring and gathering information on molecular species and their quantities that are created over the simulation (FIGURE 1.3).

1.5.1.6 Static and dynamic analysis

The Kappa framework offers graphical tools for static and dynamic analysis of a model. The former are performed without model simulation and as such are independent from kinetic parameters [157]. Static analyses provide ways for model verification without a need of simulating the model. Among the static methods are *contact maps* and *influence maps*. Contact maps produce a diagram of all agents defined in the model, and their signatures composed of binding sites and internal states. The diagram show all possible bonds between agents as defined by the rule set. Influence maps inform on potential positive and negative influences between observables and rules. For instance, a rule can have a positive influence on the other if an agent state resulting from application of the former rule can be embedded in the LHS of the latter rule [157].

The dynamic analyses are performed during the simulation. Among these are *causal flows* and *flux maps*. Causal flows show dependencies and

conflicts between rule executions for indicated observables. Flux maps inform on how much rules contribute to each others activities in terms of negative or positive contributions, and are performed per indicated time interval [157].

1.5.2 KaSim simulation method

KaSim is a specially designed simulator for the Kappa language. It is based on Gillespie's Stochastic Simulation Algorithm (SSA) [114, 119]. The Gillespie's algorithm numerically simulates time courses of molecules modelled with biochemical reaction networks [113, 119]. The algorithm uses a procedure of Continuous Time Markov Chains (CTMC) to transit between states that depend only on the current state of the system ("memoryless" Markov property) and can occur at any time point (continuous process) [107]. The Gillespie's algorithm simulates individual stochastic trajectories. These are therefore samples from deterministic evolution of probability distributions of molecular species, represented as the Chemical Master Equation (CME) [114]. CME is the most detailed representation of stochastic evaluation of reaction networks. However, its practical use is limited to a very small, usually unimolecular reactions. The Gillespie method allows to obtain results for more complex reaction networks.

An advantage of the Gillespie method over a typical procedure of solving ODEs, is that it does not approximate infinitesimal time increments by small but finite steps [114, 119]. This approximation is known to give rise to the numerical instability that causes deviation of results from the correct solution by accumulation of approximation errors [160]. Other advantage is that solution of ODEs is performed by stepping through dynamic process in a synchronised manner [120] whereas, the Gillespie's method is a concurrent and asynchronious procedure.

Growing size of reaction networks rendered Gillespie-based numerical simulations computationally intensive and efforts have been made to improve efficiency of the algorithm [161]. The standard Gillespie's algorithm requires enumeration of all the possible species and therefore, its efficiency depends on the network size that is defined by a number of reactions and molecular species [162]. This requirement of knowing all molecular species in the model precludes possibility of modelling signalling systems that dependent on activity of combinatorially complex proteins. To model these multi-state proteins,

Danos [120] proposed a generalised variant of the Gillespie's algorithm with approximation of events that may happen and a particular correction scheme [163], implemented as KaSim.

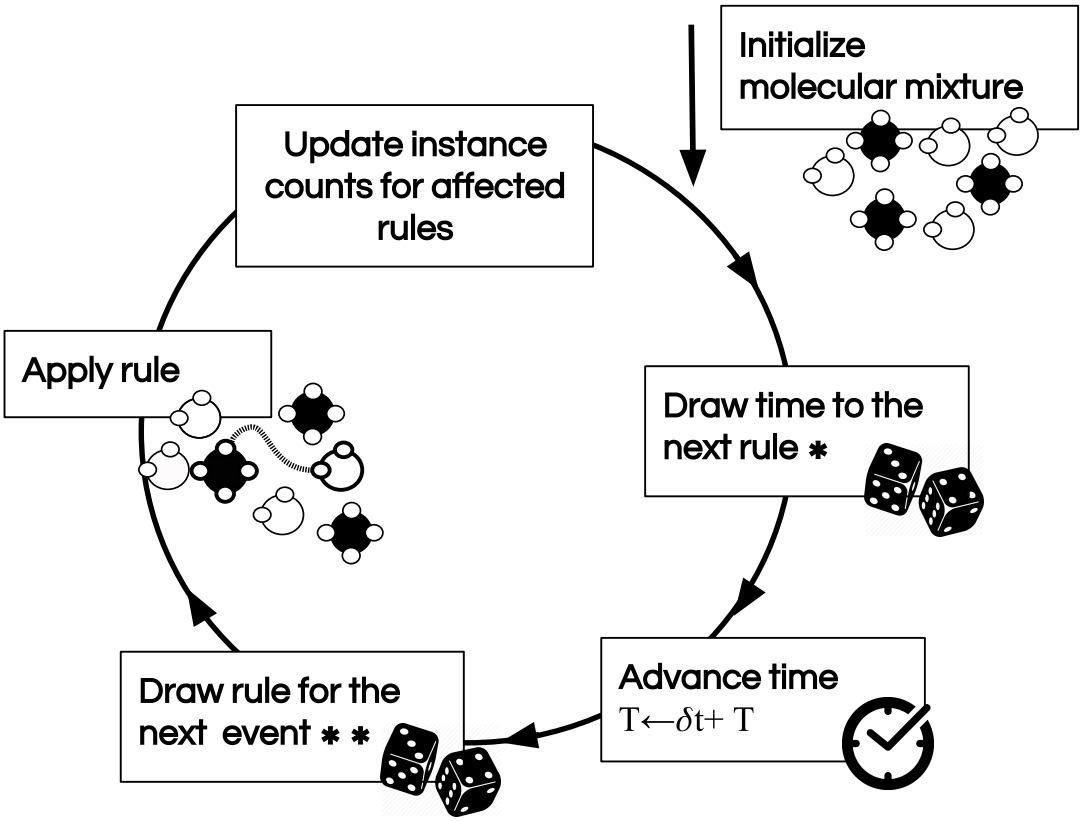
FIGURE 1.4A outlines simulation steps of the KaSim engine, presented as an *event loop* [157]. An event is application of a rule to the current mixture molecular species represented as graphs during the simulation. The procedure is based on drawing two random numbers over each event loop. First, for the time interval when the next reaction happen. Second one, for the next reaction to occur. The probability of any rule application is proportional to the rate constant of the rule, multiplied by the number of instances in the mixture that the LHS of the rule can be embedded in. This defines the rule activity (Equation 1.5). Summation of all rule activities gives the total reactivity of the system (Equation 1.6). The systems reactivity is used as a parameter to define the shape of exponential probability density function from which the next reaction event is drawn (FIGURE 1.8). In consequence, the higher the total activity of the system the shorter the time interval to the next reaction event [157].

Comparison of similar algorithms designed for network-independent simulation of RB models was performed by Suderman and Deeds [123]. The authors compared performance of KaSim and NFsim [165] simulators. NFsim is a simulator for models encoded with BioNetGen Language (BNGL). No apparent difference in efficiency and time course dynamics was observed between simulators for the same models. Thus, both simulation methods appear to be equally efficient.

1.5.3 Model examples

As a novel paradigm of dynamic modelling, the RB modelling allowed to approach experimentally general understanding and implicit assumptions regarding the nature of cell signalling and protein interactions. One of such questions concerns compositional characteristics of signalling complexes, whether they form molecular machines with well defined quaternary structure, e.g. ribosomes [122] and apoptosome [123], or whether they form pleiomorphic ensembles that are composed of dynamically changing heterogeneous complexes. Formation of pleimorphic ensembles is a hypothesis inferred from combinatorially explosion of molecular states and is yet to be experimentally

(A)



(B)

$$a(r,x) := k_r[s_r,x] \tag{1.5}$$

$$a(x) := \sum_r a(r,x) \tag{1.6}$$

$$P(r) = \frac{a(r,x)}{\sum_r a(r,x)} \tag{1.7}$$

$$P(\delta t > T) = \exp(-a(x)T) \tag{1.8}$$

FIGURE 1.4: (A) Event loop representing the KaSim engine. The next event time ("$*$") is drawn according to the probability in Equation 1.8, the probability of a rule to be executed ("$**$") is defined by Equation 1.7. (B) A list of equations defining the KaSim engine, where r is a rule, x is a state of a mixture at a time, k_r is a parameter of r, s_r is the LHS of r, $[s_r,x]$ is a number of matches of s_r in x, $a(r,x)$ is the activity of r, $a(x)$ is a total activity (or reactivity) of the system, δt is a time elapsed to the next event [164].

verified [122]. It is unknown exactly what proportion of complex-based signalling occurs through formation of molecular machines and whether the cellular signal can be reliably processed with pleiomorphic ensembles [122]. Suderman and Deeds [123] computationally examined these perspectives encoded in two RB models of Yeast pheromone pathway. In the first one, complexes were hierarchically and completely assembled to form signalling machines. In the second one, agents reacted freely forming pleiomorphic ensembles of wide variety of complex types. Phenotypic difference between models was observed, with advantage of the ensemble-based model. Unlike the machine-based one, the ensemble-based model was able to reproduce experimental observations where an overexpressed scaffold protein caused combinatorial inhibition of the model response [123].

In other study by Deeds et al. [166] explored consequences of conflicts in protein binding and combinatorial complexity in protein interaction networks. The authors created a RB model of structurally resolved interactions between Yeast proteins. Protein interaction network, that underlay the model, was enriched with information on protein binding interfaces, in particular defining which of these interactions are simultaneous and which mutually exclusive [167].

Similarly relying on resolved information on binding interfaces but through protein domain information, Sorokina et al. [168] constructed a model of postsynaptic density. The information on protein-domain architectures and domain interactions allowed to construct detailed interactions between scaffold proteins in postsynaptic density, where specification of domain-based protein interactions formed major content of the RB model [168]. Commonly analysed aspects of these three models were sizes and heterogeneity of compositions of protein complexes [123, 166, 168].

Next to these inquiries regarding general aspects of signalling protein networks, there are numerous modelling examples aiming to model specific molecular mechanisms with the RB framework. A list of models published between 2007 and 2013 is provided by Chylek et al. [169]. The authors enlisted 21 models of various topics regarding cell signalling, and 13 models of immune signalling. Summary of earlier published models than 2007 can be found in Hlavacek et al. [106]. Table 1.1 shows examples of modelled mechanisms published after 2014.

Modelled subject	Citation
Long-term potentiation and long-term depression	Antunes et al. [170]
Insulin signalling	Di Camillo et al. [171]
Lymphocyte-specific Tyrosine kinase autoregulation	Rohrs et al. [172]
Cell-line specific early signalling of EGFR	Stites et al. [173]
Base excision repair	Köhler et al. [174]
Early T-cell receptor signalling	Chylek et al. [175]

TABLE 1.1: Examples of topics modelled with the RB framework, unmentioned in the previous reviews enlisting RB models [106, 169].

1.6 Organisation of thesis

In *Chapter 2* of this thesis, I will present comparison of a RB model to an ODE model representing the same molecular system. This comparison aims to determine if dynamics encoded with the ODE model can be reproduced within the RB framework, and to establish potential advantages of modelling with rules. The RB model was obtained through translation of an existing ODE model into the RB syntax. The models are compared with respect to specification components and by overlying time courses of corresponding model outputs. The models are also analysed in different variants, a base-line and an emulation of experimental perturbation.

In *Chapter 3,* a pipeline for extended and automated analysis of RB simulation results is proposed. The pipeline is performed on the RB model presented in *Chapter 2*. In the pipeline procedure, model outputs are partitioned and scored based on sets of their time courses generated from the RB model with randomly varied parameter sets. Selected observables are passed to global sensitivity analysis (GSA) that measures sensitivity of the model output to variation of rate constants. Selected model outputs and their scored relations with parameters are represented as a weighted network graph to enhance analysis of relations between these model components. This network representation is further used to identify differences between two model phenotypes.

In *Chapter 4,* I present explorations of molecule-centred resources that are relevant in development of mechanistic and disease-related RB models. As-

semblage of such repositories could accelerate the process of model construction and direct the subject of dynamic modelling towards disease-related mechanisms. Contents and coverage of these datasets were studied with respect to Attention Deficit Hyperactivity Disorder (ADHD) as an example of complex disorder with relatively high prevalence. A list of the ADHD-associated genes was assembled from three resources representing different types of studies and qualities. These resources are examined with respect to coverage, accuracy and potential difficulties involved in their application. Among molecular-centred repositories, the main focus was laid on protein-protein interactions (PPIs), enriched with information on domain-domain interactions (DDIs) identified in the ADHD-associated proteins. Kinase-substrate interactions (KSIs) were included as most common form of PTMs that indicate reactants of phosphorylation reactions. As reactions and rate constants defining new models are often derived from existing models, the BioModels database of mathematical models is screened with the proteins associated to ADHD. Lastly, enrichment analysis is performed to identify pathways where the ADHD genes are overrepresented.

In the final *Chapter 5*, I recap discussions of results presented in the result chapters and reflect on future perspectives.

1.7 List of Acronyms

AC adenylyl cyclase

ADHD Attention Deficit Hyperactivity Disorder

ARI Adjusted Rand Index

ATP adenosine triphosphate

$\mathcal{A}$ Avogadro's constant

BioPAX Biological Pathway Exchange

BNG BioNetGen

BNGL BioNetGen Language

Ca^{2+} calcium ions

CaM calmodulin

CaMKII Ca^{2+}/calmodulin-dependent protein kinase II

cAMP cyclic adenosine monophosphate

CDK5 cyclin dependent kinase 5

CK1 casein kinase 1

CK2 casein kinase 2

CNS central nervous system

constSer137 constitutive Ser137

COPASI COmplex PAthway SImulator

CorEx Correlation Explanation

D137 DARPP-32 phosphorylated at Serine 137

D34 DARPP-32 phosphorylated at Threonine 34

D75 DARPP-32 phosphorylated at Threonine 75

DA dopamine

DARPP-32 dopamine- and cAMP-regulated neuronal phosphoprotein with molecular weight 32 kDa

DDI domain-domain interaction

DO Disease Ontology

DOQCS Database of Quantitative Cellular Signaling

DRD1 dopamine receptor D1

DRD2 dopamine receptor D2

eFAST extended Fourier Amplitude Senstivity Test

EGFR epidermal growth factor receptor

Gene ID Entrez Gene identifier

Glu glutamate

GO Gene Ontology

GPCR G-protein–coupled receptor

GSA global sensitivity analysis

GSDDI gold standard domain-domain interaction

HSIC Hilbert-Schmidt Independence Criterion

HUPO Human Proteomics Organisation

IDDI Integrated Domain-Domain Interactions

KSI kinase-substrate interaction

LSA local sensitivity analysis

LTD long-term depression

LTP long-term potentiation

MI Molecular Interactions

MIRIAM Minimal Information Requested In the Annotation of biochemical Models

MSPN medium spiny projection neurons

NCBI National Center for Biotechnology Information

NMDAR N-methyl-D-aspartate receptor

oBS one-binding-site DARPP-32

ODE ordinary differential equation

ORA over-representation analysis

PDB Protein Data Bank

PDE phosphodiesterase

PDI protein-domain interaction

PKA protein kinase A

PKAc protein kinase cAMP-activated catalytic subunit

PKAr protein kinase cAMP-dependent regulatory subunit

PP1 protein phosphatase 1 catalytic subunits

PP1 protein phosphatase 1

PP2A protein phosphatase 2

PP2B protein phosphatase 3/calcineurin

PP2C protein phosphatase 2C

PPI protein-protein interaction

PPP1R1B protein phosphatase 1 regulatory inhibitor subunit 1B isoform 1

PRCC Partial Rank Correlation Coefficient

PSI Proteomics Standards Initiative

PTM post-translational modification

R2C2 Inactive PKA in the form of a heterotetramer. It consists of two regulatory and two catalytic subunits.

RB rule-based

REACTOME REACTOME Pathway Database

RKHS Reproducing Kernel Hilbert Spaces

SA sensitivity analysis

SBML Systems Biology Markup Language

Ser102 Serine 102

Ser137 Serine 137

Ser137Ala Serine to Alanine mutation of DARPP-32 at Ser137

SNP Single Nucleotide Polymorphism

SSA Stochastic Simulation Algorithm

t-DARPP truncated DARPP

tBS three-binding-sites DARPP-32

TDCC Top-Down Coefficient of Concordance

Thr153 Threonine 153

Thr34 Threonine 34

Thr75 Threonine 75

UniProtKB the Universal Protein Resource Knowledgebase

UniProtKB AC UniProtKB accession

Chapter 2

Kappa model of DARPP-32 network

2.1 Motivations

As presented in *Chapter 1*, molecular signalling is characterised as a complex system of coupled interacting components resulting with nonadditive effects. Its understanding has been facilitated with formal methods that allow to study dynamics of such systems. A common way for studying such systems in a powerful and detailed manner is by defining molecular reactions as a set of ordinary differential equations (ODEs). However, as discussed in *Section 1.4*, mechanisms underpinning the function of large scale signalling networks demonstrates limitations of ODE-based representation. Development of modelling methods derived from computer science has specifically addressed growing complexity of represented systems. A particularly promising example of such methods is rule-based (RB) modelling, designed to model a system of interacting proteins. The potential of the method has been extensively discussed [108, 120, 124, 149] and shown on demonstrative examples [120] or with models attempting to answer new biological questions (see *Section 1.5.3*). These models were often based on existing ones, which in a great majority are defined as ODEs. However, to the author's knowledge, any direct and systematic comparison of the same biological system defined in the light of two formalisms has not been presented before. Moreover, application of an existing method [176] for automated translation of a format encoding ODE-based models to a RB model format appeared to be unsuccessful (*Appendix A*). Therefore, in this chapter, I examine differences between an existing

ODE model compared to its RB counterpart resulted from the translation of reactions underlying the ODE model to the RB syntax. With this setting, I specifically ask if dynamics of an ODE-based model can be reproduced with a RB-model? Secondly, if differences between both are observed, what are their particular underpinnings? Lastly, if dynamics of ODE-defined system are reproduced within RB one, what can be gained with a model defined in RB-setting that would extend analytical framework of the system already defined with a set of ODEs.

Such comparison of two formalisms could reveal consequences of divergence in two types of model definitions and henceforth, give a better understanding of RB modelling. For instance, as presented in *Section 1.4*, the ODE-based modelling represents a molecular system as concentrations of molecular species and focuses on their reaction kinetics. Contrary to this, RB modelling is an agent-focused method, where unfolding of molecular compositions can be studied alongside their abundances [120]. Moreover, with respect to ODE modelling, RB remains a relatively new paradigm. It still requires support in research, tool development and applications by modelling community to understand the differences in insights both formalism can offer and use it appropriately.

2.2 Introduction

To perform comparison between the two formalisms, we first need to translate an existing ODE model into a RB one. An ODE model of choice should satisfy certain requirements. The first requirement is a quality of study that is weighted by acknowledgement in the field of molecular neuroscience represented by frequency of reference to the model in successive studies. Second requirement is reproducibility criteria. To directly compare dynamics of molecular species, time-evolution of molecular concentrations has to be obtained for both models. This requires the ODE model to be encoded in a machine-readable format that could be numerically simulated with an appropriate software.

A model of the immediate interactors of dopamine- and cAMP-regulated neuronal phosphoprotein with molecular weight 32 kDa (DARPP-32) network by Fernandez et al. [177] was found to satisfy these criteria. The model is considered by the community interested in modelling dopamine-dependent synaptic plasticity to be a valuable study that can serve as a solid core for

building larger and more complex models [178]. Moreover, it is a largely cited study not only by modellers [131, 179, 180] but also experimentalists [181–183]. As for the reproducibility criteria, the Fernandez et al. [177] model was clearly created with this problem in mind. First of all, the model is encoded in the standard Systems Biology Markup Language (SBML) format, as well as the format used in its original environment (E-Cell, version 3). Furthermore, it is deposited in the BioModels database [184]. Each curated model in the database is annotated with external common identifiers complying with the Minimal Information Requested In the Annotation of biochemical Models (MIRIAM) standard [185]. For instance, proteins in a model are annotated with the standard protein identifiers of the Universal Protein Resource Knowledgebase (UniProtKB) [186] what allows for unique identification of protein sequences.

It should be noted that the translation of the model of Fernandez et al. [177] into the Kappa-based framework does not aim to demonstrate that a stochastic model is more adequate to study this particular reaction network. As copy numbers of molecules represented in the model are sufficiently high, stochastic modelling would not lead to any substantial change of conclusions obtained with the ODE-based formalism and therefore, the ODE-based model is in this particular case a benchmark solution.

In this chapter, the conversion of the ODE model to a RB model is presented in detail. First, I introduce the biological setting of the chosen system as well as the potential advantages of translating this system into a RB model. Then I cover aspects of the model translation, such as how agents and observables where defined as well as the reaction(s)-to-rule(s) conversions. I present the results of model comparison at the notation level and using model dynamics under different conditions. Following that, I discuss the advantages and disadvantages of the two model representations, and present suggestions for future work.

2.2.1 Role and importance of the DARPP-32 protein

DARPP-32, officially named protein phosphatase 1 regulatory inhibitor subunit 1B isoform 1 (PPP1R1B) [187], is an important multistate and intrinsically disordered protein [188] regulating synaptic plasticity. DARPP-32 was first discovered by Walaas et al. [189]. Although expressed in multiple brain

regions [190], it is a central protein studied in dopamine (DA)-dependent plasticity in medium spiny projection neurons (MSPN) of striatum.

Striatum is a subcortical brain structure and the largest nuclei of basal ganglia. Striatum integrates multiple inputs to the basal ganglia circuit, such as glutamatergic excitatory afferents from the cortex and dopaminergic inputs from the midbrain [191]. Around 95% of Human striatal cells are MSPNs, in which signalling cascades activated simultaneously by glutamatergic and dopaminergic stimuli is a necessary condition for the long-term potentiation (LTP) that underlies context and reward-related learning [192].

The importance of studies on DARPP-32 in striatal signalling underpinning DA-dependent synaptic plasticity stands in analogy with Ca^{2+}/calmodulin-dependent protein kinase II (CaMKII) in models of Ca^{2+}-dependent synaptic plasticity in hippocampal neurons [38]. Synapses in striatal MSPNs share some pathways with hippocampal synapses that carry glutamate (Glu)-induced signal and include striatum-specific proteins involved in the DA signal. These two signals are integrated by DARPP-32 that is involved in a complex network of interactions regulating its phosphorylation sites. There are at least eight phosphorylation sites in the DARPP-32 amino acid sequence that have been confirmed by multiple studies[1]. However, only four are known to have a regulatory impact on DARPP-32 [193]. These four phosphorylation sites are Threonine 34 (Thr34), Threonine 75 (Thr75), Serine 137 (Ser137) and Serine 102 (Ser102) (positioned on the *Rattus norvegicus* protein sequence). The multiplicity of phosphorylation states leads to a large number of interacting partners of DARPP-32 that affect these states. Based on the general process of phosphorylation, these interactors can be divided into two types, protein kinases and phosphatases, which are important signalling modulators that mediate phosphorylation and dephosphorylation, respectively [1].

Among DARPP-32 interactors we can find protein phosphatase 2 (PP2A), protein phosphatase 3/calcineurin (PP2B), protein phosphatase 1 (PP1), protein phosphatase 2C (PP2C), cyclin dependent kinase 5 (CDK5), protein kinase A (PKA), casein kinase 1 (CK1) and casein kinase 2 (CK2). The first three phosphatases are multimers composed of functionally different subunits. PP2C denotes a protein family of monomeric enzymes. PKA is a multimeric protein

[1] *Homo sapiens* protein sequence: https://www.ncbi.nlm.nih.gov/protein/NP_115568.2. Accessed 04-07-2017.

family of kinases. The last two Caseins are distinct protein families.

The Threonine sites (Thr34, Thr75) have major regulatory roles in signal processing. The former inhibits PP1 and the latter inhibits PKA, which in turn phosphorylates Thr34. On the other hand, PKA activates the phosphatase of Thr75, PP2A. The Serine sites (Ser137, Ser102) have a supporting role in Thr34 signal enhancement. Ser137 inhibits dephosphorylation of Thr34 by PP2B and Ser102 increases phosphorylation of Thr34.

It has been shown that DARPP-32 malfunction and abundances relates to multiple central nervous system (CNS) disorders. Among these are Alzheimer disease [194], addiction [195], affective disorders [196] and schizophrenia [196, 197]. Its malfunction has been associated with missing phosphorylation sites that define its function. The lack of phosphorylation site can be caused by splice variation, e.g. truncated DARPP (t-DARPP) [196] or protein cleavage [194]. The former lacks Thr34 (the *Rattus norvegicus* protein sequence) and the latter is cleaved at Threonine 153 (Thr153) (the *Mus musculus* protein sequence). Both of these mutations impair the PP1 inhibitory function of DARPP-32. DARPP-32 is also indirectly linked to multiple diseases through its mediation of symptoms caused by psychoactive drugs that affect DA transmission [193]. It has even been proposed as a potential drug target for dopamine-related disorders [198].

The study of Stipanovich et al. [199] showed that nuclear accumulation of DARPP-32 is promoted by drugs of abuse. They found that nuclear DARPP-32 is essential for gene expression via phosphorylation of histone H3. The nuclear transportation of DARPP-32 is regulated by Ser102 phoshophorylated by CK2.

2.2.2 Advances in DARPP-32 network modelling

An early dynamic model of DA-dependent synaptic plasticity that contains DARPP-32 was built by Kötter [200]. The major aim of Kötter [200] was to study molecular mechanisms underlying interactions between DA and Glu afferent signalling in MSPNs. As the very first computational approach to this problem, the model was represented as a set of equilibrium equations. The major results report the quantitative sensitivities of phosphatases and kinases to Glu and DA-signalling, dividing it into sensitive to the paired signals or to Glu alone.

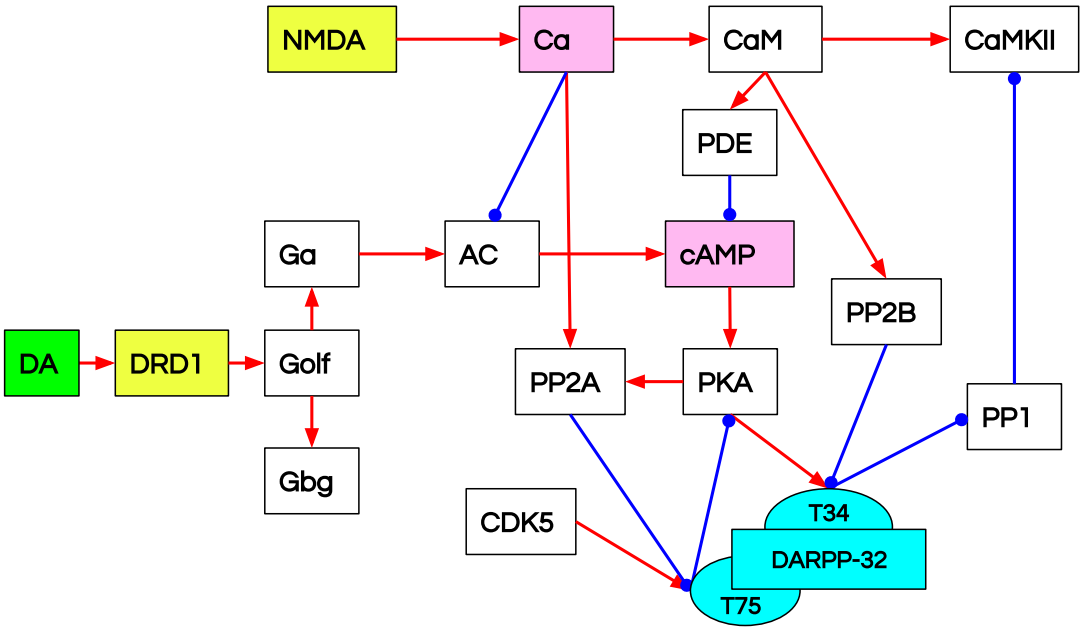


FIGURE 2.1: Diagrammatic overview of the Lindskog et al. [201] model of glutamatergic and dopaminergic signal integration in MSPNs.

The first ODE-based models of the DARPP-32 network were built in the early 2000s by Lindskog et al. [201] and Fernandez et al. [177]. Similarly to the model of Kötter [200], both models focused on Glu and DA signal integration as an important factor of synaptic plasticity enhancing connections between neurons of striatum.

Lindskog et al. [201] created a model of interacting cascades activated by DA and Glu signalling that stimulated the dopamine receptor D1 (DRD1) and influx of calcium ions (Ca^{2+}) through the N-methyl-D-aspartate receptor (NMDAR), respectively (FIGURE 2.1). The glutamatergic signalling cascade included the CaMKII protein activation by calmodulin (CaM). The dopaminergic signalling started with DA that bound postsynaptically to DRD1, that as a G-protein–coupled receptor (GPCR), caused the G protein to dissassociate G_α and $G_{\beta\gamma}$ subunits. Subsequently, G_α activated adenylyl cyclase (AC), that catalysed production of cyclic adenosine monophosphate (cAMP) from adenosine triphosphate (ATP). cAMP activated PKA, that phosphorylated DARPP-32 at Thr34. DARPP-32 phosphorylated at Thr34 inhibited PP1, that dephosphorylates CaMKII.

In the model Thr34 is both activated and inhibited by the Ca^{2+} feed-forward signal, that is conveyed by the PKA–PP2A–Thr75 double negative

feedback loop. PP2B dephosphorylates Thr75 but its action is enhanced by Ca^{2+} and PKA. The model showed that the loop does not exclusively reinforce PKA pathway stimulated by DA but instead acts as a competitive inhibitor for PKA.

Lindskog et al.'s 2006 model become a foundation of a number of models focused on different aspects of DA and Glu signalling integration, developed in the Hellgren-Kotaleski Laboratory [179, 202, 203] and in other laboratories [204, 205]. These models only included Thr34 and Thr75 as major switching factors between LTP and long-term depression (LTD). Since then, several models of the system were built reusing all or some part of these three models and extending them with downstream and/or upstream signalling events [180, 202–205].

All these dynamic models differ in the number of DARPP-32 phosphorylation sites included in models. Fernandez et al. [177] and Nakano et al. [180] included three phosphorylation sites. Barbano et al. [206] and Qi et al. [204] included four sites. These models incorporating all four phosphorylation sites, were mainly focused on the overall system response either as a test case for a novel method to analyse system robustness [206], or to test variable scenarios of input signals [204]. All four phosphorylation sites have rarely been studied in models due to the combinatorial complexity involved in modelling all combinations of DARPP-32 states, that was claimed as unnecessary with respect to the studied subject [201]. For instance, a number of models of the DA and Ca^{2+} signal integration have included only Thr34 and Thr75 as major switching factors between LTP and LTD [201–203, 205]. As all models of DARPP-32 were built with ODEs, extending such models with the additional molecular species would cause growth in complexity of model specification. Addition of new reactions might not only require specification of additional kinetic laws but also rewriting the existing equations if concentrations of exiting molecular species were affected by the added ones. This could be the least problematic task if the enumeration of all possible molecular species was realistic to enumerate (see an example of EGFR in *Section 1.1*). Therefore, if a large number of molecular species is already included in an existing model, addition of new reactions and molecular entities might become more difficult and laborious as the complexity of the model increases. What often is practised in such occasions is omission of molecular species and reactions which are assumed as non-contributing or not affecting the subject of inquiry, or aggregation (*lumping*) of molecular species

into one constituent. A simplified or reduced model might not align with the future aims and discoveries made on the studied system. It might be argued that reusing such reduced models as modules could be difficult. For instance, so far only early signalling events of DARPP-32 signalling has been modelled, localised mainly in cytosol. Therefore, Ser102 was omitted in the model because of no evidence that it can be affected by DA or Glu signalling [207, 208]. However, a recent study by Nishi et al. [17] suggests that Glu can decrease the effect of DA signalling (phosphorylation of three other sites) by dephosphorylating DARPP-32 at Ser102 that causes accumulation of DARPP-32 in nucleus.

Since DARPP-32 was first discovered by Walaas et al. [189], there has been extensive research on its mechanisms of action driving multiple hypothesis of its general role in the CNS [208]. However, most of these emerging hypotheses have not been modelled or tested in integrated, formal representations. Moreover, ongoing research around DARPP-32 implies also need of constant development of new models of DARPP-32-involved processes. The traditional mode of constructing bespoke models does not help to tighten the connection between *in silico* modelling and recent discoveries in molecular biochemistry.

2.2.3 The Fernandez model of DARPP-32 signalling

Fernandez et al. [177] is a study of the integrative effect and sensitivities of DA- and Glu-mediated signals on the DARPP-32 network in striatal DRD1-expressing neurons. The model examined the effect of cAMP-pulse followed by Ca^{2+} spike trains varying the distance between the stimuli. The study showed that DARPP-32 is a robust integrator, indifferent both to its initial concentration and delay between the stimuli.

At the time of the model creation, it was suggested that DARPP-32 was a bistable-switch between DA and Glu signals, with cAMP and Ca^{2+} as second messengers, respectively. The former inhibits PP1 when DARPP-32 is phosphorylated at Threonine 34 (Thr34). The latter inhibits PKA when DARPP-32 is phosphorylated at Threonine 75 (Thr75) counteracting DA-triggered events. Further studies showed that the phosphorylation patterns activated by the Glu input are far more ambiguous and complex. The study of Fernandez et al. [177] addressed some of these complexities.

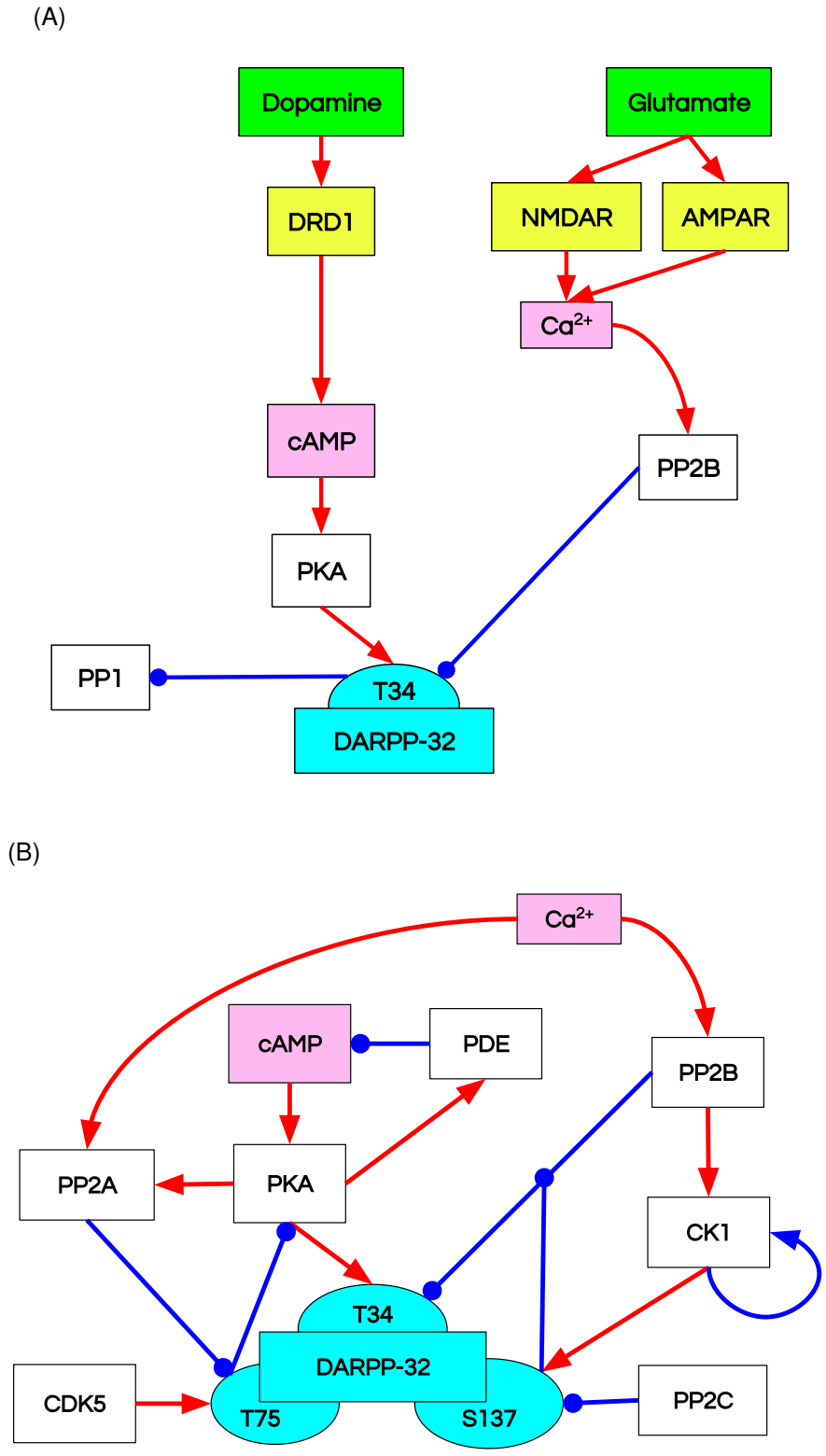


FIGURE 2.2: Reaction diagrams representing different aspects of the DARPP-32 network: (A) simplified with the emphasis on incoming signal and receptors, which are not included in the model; (B) included in the ODE model by Fernandez et al. [177]. Nodes: stimuli *(green)*, receptors *(yellow)*, second messengers *(magenta)*, kinases/phosphatases *(white)*. Edges: inhibiting reactions *(blue)*, activating reactions *(red)*.

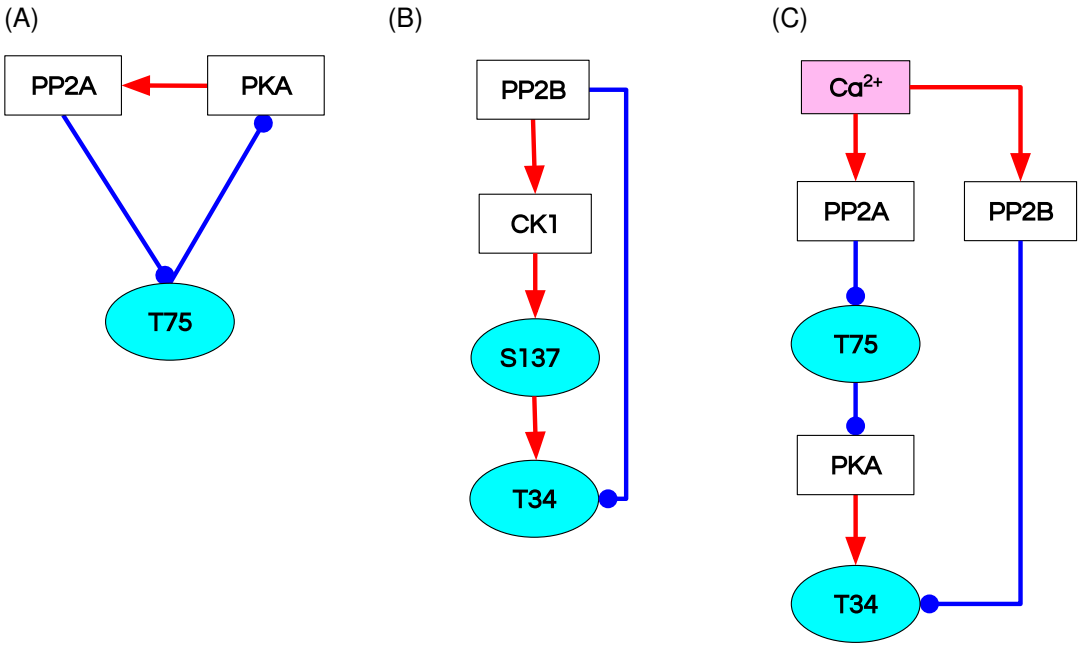


FIGURE 2.3: Reaction diagrams representing DARPP-32 subnetworks included in the model: (A) double negative (positive) feedback loop (B) & (C): incoherent feedforward loops. Nodes: phosphorylation sites of DARPP-32 *(cyan)*, second messengers *(magenta)*, kinases/phosphatases *(white)*. Edges: inhibiting reactions *(blue)*, activating reactions *(red)*.

To reproduce the system's known behaviour, the authors included two main pathways that mediate these signals, DA–cAMP–PKA–DARPP-32 phosphorylated at Threonine 34 (D34) and Glu–Ca^{2+}–PP2B–DARPP-32 phosphorylated at Threonine 75 (D75) (Figure 2.2A).

Contrary to the model of Lindskog et al. [201] that also investigated integration of Glu and DA signals, the reaction network consists of DARPP-32 with three phosphorylation sites: Thr34, Thr75 and Ser137. Each of these sites has an associated kinase and phosphatase: PKA and PP2B (Thr34); CDK5 and PP2A (Thr75); CK1 and PP2C (Ser137).

As mentioned earlier, the fourth phosphorylation site, Ser102, was not included. The decision to exclude Ser102 from the model was based on its weak effect on the phosphorylation of Thr34 site by PKA. Moreover, neither of the stimuli was known to regulate Ser102's phosphorylation.

In the model, the authors included two incoherent feedforward loops triggered by the Ca^{2+} influx that both activates and inhibits Thr34. The first one activates Thr34's protein phosphatase, PP2B, that disinhibits CK1. This in turn phosphorylates Ser137, known to inhibit dephosphorylation of Thr34 by PP2B (Figure 2.3B). The second incoherent loop was induced by the enhancement of PP2A dephosphorylation of tonically active D75 by its complexation with Ca^{2+} (Figure 2.3C). Also, the phosphorylation of PP2A by PKA, activated on cAMP signal, enhances the dephosphorylation of D75. The second incoherent feedforward loop was designed to test the mechanism of signal integration based on the experimentally observed decrease of both Thr34 and Thr75 on Ca^{2+} stimulus [209]. The effect of this mechanism was tested by comparing two models, "model A", without the mechanism, and "model B", with the mechanism. "Model B" was represented with 32 supplementary reactions added to "model A". "Model B" showed a decreasing effect of Thr75 triggered by Ca^{2+} influx. The results matched the experimental findings but did not affect substantially PKA or Thr34 activity. Nevertheless, the simulation results showed that DARPP-32 is not a bistable switch. "Model B" is used in this study as this model variant reflected experimental results. For a detailed comparison of mechanisms encoded in the two models see the original study of Fernandez et al. [177].

Another important role of PP2A tested in the model was the PKA–PP2A–D75 double negative feedback loop (Figure 2.3A). It was shown that the

loop does not exclusively reinforce the PKA pathway activated by DA but acts as a competitive inhibitor for PKA that balances D34 activation.

The focus of the model was on the role and sensitivities of the phosphatases and kinases of DARPP-32 to Ca^{2+} and cAMP-signals. Therefore, although upstream events to direct DARPP-32 interactions were known at the time, they were omitted or fused into simplified representations. For instance, the Glu-activated cascade, including the CaMKII circuit, is represented by direct Ca^{2+} binding to the phosphatases and kinases that act on DARPP-32. As AC was not included in the model, the inhibitory effect of the Glu signal was also absent from the model. Similarly to Glu-activated cascade, the pathway directly activated by DA was also omitted by abridging the G-protein signalling and the cAMP production step with a direct "injection" of cAMP molecules during the simulation.

The authors also tested the role of Ser137 in model dynamics with two *in silico* mutagenesis modifying its function. These mutations have two oppose effects on Ser137. The first one inhibits the phosphorylation. In the experimental perspective, it can be compared to Serine to Alanine mutation of DARPP-32 at Ser137 (Ser137Ala), where Serine at 137 position is mutated to Alanine. It was achieved by setting all catalytic constants to 0 in phosphorylation reactions of DARPP-32 at Ser137 induced by kinase CK1. In consequence, CK1 can bind to DARPP-32 but the site cannot be phosphorylated. This mutation enhances the effect of Ca^{2+} stimuli on the Thr34 site phosphorylation manifested in its greater drop in concentration after the Ca^{2+} stimulus than in the unmodified model.

The second mutation leads to indefinite phosphorylation, where the Ser137 site is always phosphorylated (constSer137). It was set by changing all catalytic constants to 0 in dephosphorylation reactions of DARPP-32 at Ser137 induced by phosphatase PP2C. As Ser137 inhibits dephosphorylation of Thr34, the mutation resulted in sustained effect of the cAMP stimulus and little impact of the Ca^{2+} stimulus on Thr34 dephosphorylation.

2.3 Methodology

The aim of this chapter is describe the set of procedures to perform a direct comparison of RB and ODE frameworks. Figure 2.4 shows a brief outline of the approach. To obtain two representation of the same molecular

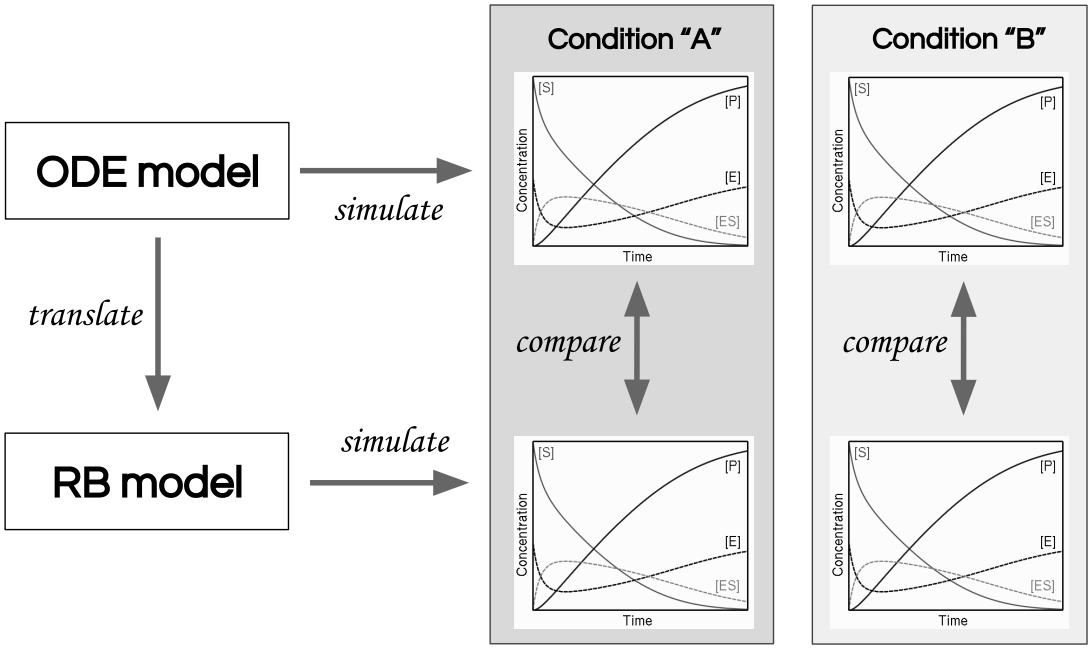


FIGURE 2.4: Approach to methodology for comparison of ODE and RB modelling frameworks. Source of graph plots: Wikipedia

system, in the first step, reactions underlying the ODE model of Fernandez et al. [177] are translated to a RB language. In the next step, both models are simulated with variable modifications to obtain time courses of selected observables. Model translation can be divided into components that consists of definitions of molecular agents, translation of rate constants and molecular abundances from concentrations to copy numbers, encoding of stimuli during the simulation and the simulation settings. In the next part of this section, procedures required to compare time courses obtained with simulations are described. Two variants of modifications applied to models' baseline settings are presented as well as details regarding applied simulators and a method of selecting equivalent observables.

2.3.1 Model translation

In order to translate the model from ODE to RB, reactions underlying the set of equations are *decontextualised*. This involves identification of agents that have persistent molecular identities and generalisation of reactions into reaction patterns that contain only relevant information denoting necessary conditions for reaction to occur. We also need to translate molecular concentrations, rate constants and initial molecular abundances to copy-numbers. Lastly,

the cAMP pulse and the Ca^{2+} spiking, specified as events in the original model, are translated into the RB syntax as a set of modification rules executed during the simulation.

2.3.1.1 Definition of agents

Of 75 molecular species in the model, 11 were identified as agents, based on their persisting chemical identity as described in *Section 1.5.1.1*. Agent's signatures were defined based on internal states and binding capabilities contained in reaction definitions. Signatures of agents with more complex structures are described here in more detail. These are DARPP-32, PP2A, PP2B and R2C2, a heterotetramer harbouring inactive PKA.

The signature of DARPP-32 is defined with one binding site and three internal states, each with phosphorylated and unphosphorylated states. The binding site has multiple binding partners, which are the kinases and phosphatases of the three phosphorylated sites.

PP2A was defined as having two binding sites, one for Ca^{2+} and the other for PKA and D75. The latter binding site has two states, phosphorylated and unphosphorylated. This particular design is implemented under two assumptions. First, the phosphorylation state and binding to Ca^{2+} are independent of binding to PKA or D75. Secondly, PP2A exclusively binds to either PKA or D75.

PP2B is activated if bound to four Ca^{2+} ions. Hence, its signature was defined with four Ca^{2+} binding sites and a separate site reserved for an internal state description, either "active" or "inactive". As in the ODE model, PP2B binds to two Ca^{2+} ions at a time.

A kinases phosphorylating DARPP-32 at the Thr34 site, denoted as PKA, is in fact protein kinase cAMP-activated catalytic subunit (PKAc), which in the inactive form is a part of heterotetramer R2C2. It consists of two regulatory and two catalytic subunits, protein kinase cAMP-activated catalytic subunits (PKAcs) and protein kinase cAMP-dependent regulatory subunits (PKArs). PKAc is activated by dissociation from regulatory subunits by binding of four cAMP to R2C2. This mechanism was encoded by the introduction of an R2C2 agent with four binding sites for cAMP and two internal states, "on" and "off", denoting disassociation of two catalytic subunits, further called as PKA for consistency with the original model. If all four binding sites are occupied,

then one of the two states turns "on" and one PKA agent is created during the simulation.

2.3.1.2 Translating reactions into rules

The rules of the RB model were encoded based on the list of reactions published in Fernandez et al. [177] as "model B". It is composed of 152 elementary irreversible chemical reactions that are combined with the same number of rate constants to define rate laws. All reactions can be divided into three classes: binding, phosphorylation and creation. Each of these reaction classes is complemented with counteracting reactions: dissociation, dephosphorylation and degradation. The phosphorylation and dephosphorylation occurs in two steps. First, a kinase or phosphatase binds to its substrate, then it dissociates with or without the site phosphorylated or dephosphorylated. It is assumed that kinase or phosphatase can bind to its substrate on the condition that the substrate is not already phosphorylated or dephosphorylated at the target site. This assumption is called as absence of product rebinding [177].

As explained in *Section 1.5.1.2* of the introductory chapter, a reaction can be written as a rule and therefore, the translation could have been accomplished in a one-to-one manner. However, to fully take advantage of rule patterns, multiple reactions can be condensed into fewer rules by removing irrelevant context, i.e. *decontextualised*. The context of a reaction in the RB model is defined as the information about the agent's binding sites, partners and internal states. Based on this definition of reaction context, the following criteria guided decisions about condensing reactions into rules. Given a set of reactions of the same type, either forward, backward, or catalytic, between two the same reactants (agents), if the difference between reactions lays in agent states, either internal or binding, that does not change after the transition from reactants to products, and reaction constants in all these reactions have the same values, then information about agent states is redundant to define reaction conditions; hence, they can be removed from the reaction notation and form a single rule pattern. For instance, the following reactions represent binding of DARPP-32

by CDK5, that in the next step catalyses phosphorylation of Thr75:

$$\mathrm{D} + \mathrm{CDK5} \xrightarrow{\mathrm{kon1}} \mathrm{D{-}CDK5} \tag{2.1}$$

$$\mathrm{D34} + \mathrm{CDK5} \xrightarrow{\mathrm{kon4}} \mathrm{D34{-}CDK5} \tag{2.2}$$

$$\mathrm{D137} + \mathrm{CDK5} \xrightarrow{\mathrm{kon11}} \mathrm{D137{-}CDK5} \tag{2.3}$$

$$\mathrm{D34{:}137} + \mathrm{CDK5} \xrightarrow{\mathrm{kon18}} \mathrm{D34{:}137{-}CDK5} \tag{2.4}$$

To represent binding of CDK5 to DARPP-32, reactions for four different states of DARPP-32 are specified: unphosphorylated DARPP-32 (“D”), DARPP-32 phosphorylated only at Thr34 (“D34”), DARPP-32 phosphorylated only at Ser137, and DARPP-32 phosphorylated at Thr34 and Ser137 (“D34:137”). Because of the absence of product rebinding, CDK5 binds to DARPP-32 only when the Thr75 site is unphosphorylated. Therefore, combinations of states with phosphorylation of Thr75 are not mentioned in the reaction set. To indicate that an enzyme could bind to the product of its activation, these four reaction have to be complimented with another four reactions between CDK5 and four combinations of DARPP-32 phosphorylated at Thr75. The rate constants, written above each reaction arrow (“kon1”,“kon4”, “kon11”, “kon18”), are the same for all four reactions. With earlier mentioned translation criteria, this set of reactions can be translated to a rule as in Code 2.1.

Code 2.1: Rule representation of CDK5 binding to D75 (no product rebinding)

```
1 D(s, thr75~u), CDK5(a) -> D(s!1, thr75~u), CDK5(a!1)  @'kon1'
```

The full signature of agent representing DARPP-32 is:

```
1 %agent: D(s, thr34~u~p, thr75~u~p, ser137~u~p)
```

The binding site is denoted as “s”. Three phosphorylation sites as “thr34”, “thr75” and “ser137”, that can be in either of two states: “~u” or “~p”. In the rule specified in Code 2.1, the “thr75” site is explicitly mentioned to be in the unphosphorylated state when binding to “CDK5” because of the absence of product rebinding. Because all rate constants are the same for four possible states of DARPP-32 when CDK5 binds to it, the actual binding reaction is independent of the states of two remaining sites, and therefore, the rule in Code 2.1 represents all four reactions. There are exactly two elements of context that condition reaction execution:

- binding sites of interactors are free
- "thr75" site is in state "u"

Removing the second condition would mean that the product can rebind to its enzyme. Encoding of such assumption, showed in Code 2.2, is the most generic rule for binding of CDK5 to DARPP-32.

Code 2.2: Rule representation of CDK5 binding to D75 (product rebinding)

```
D(s), CDK5(a) -> D(s!1), CDK5(a!1)  @'kon1'
```

The above example describes a case where phosphorylation sites are independent of each other, that is their states do not inhibit nor change reaction rates. In the set of reactions describing the model of Fernandez et al. [177] not all phosphorylation sites are independent of each other. These dependences can be first observed in the variation of values of constant rates assigned to each reaction. In the rule notation, it results in more detailed context definition for a reaction, i.e. more precision in the rule specification. For instance, since all variations of DARPP-32 phosphorylated at Thr75 inhibit phosphorylation of Thr34 by PKA, the state of Thr75 has to be explicitly mentioned in the rule specification. Rules defined in Code 2.3 determine necessary conditions for the Thr34 phosphorylation. The first rule alone is sufficient to describe this behaviour; however, to allow for future exploration of the Thr75 inhibition of PKA as encoded in the ODE model, the second rule is also included in the RB model. In the baseline setting, the catalytic constant of the second rule ("kcat8") is set to zero and therefore, this rule will not be selected for execution as its activity is also equal to zero (*Section 1.5.2*).

Code 2.3: D34 phophorylation rules

```
   D(s!1, thr34~u, thr75~u), PKA(a!1)
-> D(s  , thr34~p, thr75~u), PKA(a  )   @'kcat3'

   D(s!1, thr34~u, thr75~p), PKA(a!1)
-> D(s  , thr34~p, thr75~p), PKA(a  )   @'kcat8'
```

There are other examples of such influence that phosphorylation sites have on each other. For instance, D75 is a poor substrate for PP2B, that is encoded as an increase in the *off*-rate dissociating PP2B from DARPP-32, whenever Thr75 is phosphorylated. The next example is DARPP-32 phosphorylated

at Serine 137 (D137) that is also a poor substrate for PP2B. This fact is encoded as a decrease in all rates of Thr34 dephosphorylation by PP2B if Ser137 is phosphorylated.

All above mentioned rules represent the first binding step of the two leading to substrate phosphorylation that, in this example, activates one of the sites of DARPP-32. It is the simplest activation scheme in the reaction set. There are also cases of more complex substrate activations, where multiple molecules of the same type bind substrates on multiple sites, further called as *combinatorial binding*. This require a particular approach, applied after Danos et al. [210], as the Kappa syntax explicitly defines binding sites of interactors where each binding site has to be unique and therefore, all binding combinations had to be explicitly encoded.

The PKA and PP2B activation schemes are two such complex cases. PKA activation is encoded as a combinatorial binding of four cAMP molecules to four identical sites of R2C2 that dissociates into two catalytic subunits. Moreover, only one cAMP molecule can bind at a time. Similarly, PP2B activation requires four Ca^{2+} ions to bind. However, because two Ca^{2+} ions bind to PP2B at a time, as specified by the original model, its activation required less binding combinations to be represented as rules. As an example of combinatorial binding, we can compare one of four PP2B activation represented as reactions is defined as follows:

$$\mathrm{PP2Bi} + 2\,\mathrm{Ca} \longrightarrow \mathrm{PP2BiCa2} \tag{2.5}$$

This reaction is encoded as six rules with Code 2.4. Each rule is written in two rows with the left hand site of the rule in odd numbered rows and the right hand site in even numbered rows starting with an arrow (->). All possible bindings of two Ca^{2+} ions to two of four PP2B sites are explicitly defined. The same method of notation is used for PKA activation reactions with the difference that only one cAMP molecule can bind at a time.

Code 2.4: PP2B activation - first reaction

```
1    PP2B(ca1  ,ca2  ,ca3  ,ca4  ,state~i),Ca2+(a  ),Ca2+(a  )
2 -> PP2B(ca1!1,ca2!2,ca3  ,ca4  ,state~i),Ca2+(a!1),Ca2+(a!2)
3
4    PP2B(ca1  ,ca2  ,ca3  ,ca4  ,state~i),Ca2+(a  ),Ca2+(a  )
5 -> PP2B(ca1!1,ca2  ,ca3!2,ca4  ,state~i),Ca2+(a!1),Ca2+(a!2)
```

```
6
7    PP2B(ca1   ,ca2   ,ca3   ,ca4   ,state~i),Ca2+(a   ),Ca2+(a   )
8  -> PP2B(ca1!1,ca2   ,ca3   ,ca4!2,state~i),Ca2+(a!1),Ca2+(a!2)
9
10   PP2B(ca1   ,ca2   ,ca3   ,ca4   ,state~i),Ca2+(a   ),Ca2+(a   )
11 -> PP2B(ca1   ,ca2!1,ca3!2,ca4   ,state~i),Ca2+(a!1),Ca2+(a!2)
12
13   PP2B(ca1   ,ca2   ,ca3   ,ca4   ,state~i),Ca2+(a   ),Ca2+(a   )
14 -> PP2B(ca1   ,ca2!1,ca3   ,ca4!2,state~i),Ca2+(a!1),Ca2+(a!2)
15
16   PP2B(ca1   ,ca2   ,ca3   ,ca4   ,state~i),Ca2+(a   ),Ca2+(a   )
17 -> PP2B(ca1   ,ca2   ,ca3!1,ca4!2,state~i),Ca2+(a!1),Ca2+(a!2)
```

2.3.1.3 Translation of concentrations to copy-numbers

In deterministic ODE-based models measured units are concentration-based, denoting molecular abundance per unit volume. RB models are defined in a stochastic framework and therefore, they are based on the numbers of molecules [119]. For instance, the reaction rate in deterministic models is expressed in M/s, whereas the reaction rate in stochastic models is s^{-1}. Therefore, the units of the ODE model can be related to the RB model by a factor of volume (V) and Avogadro's constant ($\mathcal{A}$). In the Fernandez et al. [177] model, V was equal to 1×10^{-15} l, denoting the volume of a dendritic spine compartment. Avogadro's constant is defined as the number of particles within a mol of substance, equal to $\mathcal{A} = 6.022\,141\,29 \times 10^{23}\,\mathrm{mol}^{-1}$.

All conversions were performed as described in the Kappa manual [157] and a primer for rule-based modelling [154]. The following paragraphs present detail of the conversion of molecular abundances and rate constants.

Except for this standard approach to translation of deterministic to stochastic units, other modifications of parameters aiming to match the RB model to the ODE one were not attempted. This decision was dictated by the aim to expose differences between rules and reactions defining the ODE model with the same parameter values. By preserving the original parameters, particular groups of rules that require adjustments regarding rate constants are exposed and discussed.

Conversion of rate constants In the reaction rate equations, the rate constant is a constant of proportionality that relates reaction rates to reactant concentrations that are raised to the power of their respective reaction orders. The order of reaction, defined per reactant or product, is an experimentally defined exponent. This formulates a reaction law (Equation 2.8). A deterministic reaction rate is defined as a change of initial species concentrations per time unit (Equation 2.7), which in the stochastic framework is understood as a reaction probability per time unit [119]. The reaction law for an elementary reaction in the deterministic framework is based on the law of mass action, which also applies directly to the stochastic reaction rates. The law generally states that the reaction rate is proportional to the product of reactant concentrations [105, p.141].

Given a following reaction

$$\mathrm{aA} + \mathrm{bB} \longrightarrow \mathrm{cC} \tag{2.6}$$

A, B and C are reactants and a product of the reaction, and a, b and c are stoichiometric coefficients, the deterministic reaction rate formulates as

$$r = -\frac{1}{a}\frac{d[\mathrm{A}]}{dt} = -\frac{1}{b}\frac{d[\mathrm{B}]}{dt} = \frac{1}{c}\frac{d[\mathrm{C}]}{dt} \tag{2.7}$$

that is equal to the following rate law:

$$r = k[\mathrm{A}]^x[\mathrm{B}]^y \tag{2.8}$$

The k is the concentration based reaction constant. The exponents x and y will both take values of 1, resulting in the total reaction order of 2.

In an elementary reaction, a partial order of reaction with respect to a reactant is equal to a stoichiometric coefficient of the reactant. An elementary reaction is a single step and single transition reaction that is defined with no intermediates on a molecular level [211]. The overall order of an elementary reaction is the same as its molecularity. Molecularity of a reaction is the number of reacting molecules and is deduced from a balance equation. The order of reaction, stoichiometric constant and molecularity have no relation if a reaction occurs in multiple steps [211]. In the Fernandez et al. [177] model, all enzymatic processes where decomposed into three elementary reaction steps. Based on this assumption, the units of reaction constants in the Fernandez et al.

[177] model were defined. It is an important note as the units of deterministic rate constants depend on the total reaction order, that also determines the conversion to stochastic rates. The unimolecular reactions are first-order with units of s^{-1}. These are reactions of dissociation, phosphorylation and dephosphorylation. Bimolecular reactions, mostly formalising binding reactions, are second-order with units $M^{-1}.s^{-1}$. Two reactions are trimolecular, where two ions of Ca^{2+} bind to one molecule of PP2B, free or occupied with other Ca^{2+} ion. This is a third order reaction with units $M^{-2}.s^{-1}$. There is one zero-order reaction for Ca^{2+} influx with units $M.s^{-1}$.

To translate k to its stochastic equivalent k', the following equation was applied:

$$k' = \frac{k}{(\mathcal{A}\, V)^{(n-1)}} \tag{2.9}$$

where V is the volume, $\mathcal{A}$ is the Avogadro's number, $n \geq 0$ is a total reaction order.

In the original model, around 65% of reaction constants were estimated to match concentrations of different phosphorylation states of DARPP-32 observed *in vivo* at the steady state and reviewed by Svenningsson et al. [208] [177]. The remaining 35% are derived from the Database of Quantitative Cellular Signaling (DOQCS) [212], BRENDA [213] and two published models of hippocampal [214] and cerebellar brain regions [215].

Conversion of molecular species abundances The conversion of molecular abundances is necessary to specify initial molecule states. In deterministic models they are defined in mole/l (M). To translate the concentration to the number of molecules, the concentration is multiplied by V and $\mathcal{A}$ in l/mol units [157]. However, because all the initial states were provided in the Fernandez et al. [177] paper in both molecular numbers and concentrations, there was no need to apply this rule.

2.3.1.4 Stimuli application

Modifications are designed to reproduce the stimulus of the original model. It includes two stimuli during the course of the simulations. These are sudden and large increases of cAMP and Ca^{2+} (Figure 2.5).

The cAMP pulse is followed by ten Ca^{2+} spike trains after 50 seconds. Ca^{2+} is introduced to the system with a constant influx and outflux of Ca^{2+}

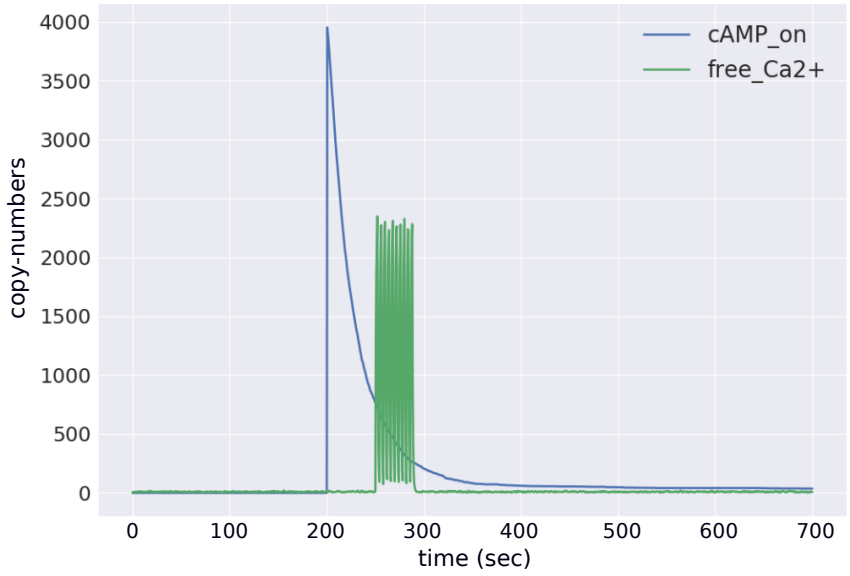


FIGURE 2.5: During the course of simulation of the DARPP-32 network model, two different types of stimuli are introduced. These are a cAMP pulse and 10 Ca^{2+} spikes. The first one occurs after 200 seconds and the second one after 250 seconds.

from the onset of the simulation. The induction of the spike train in the ODE model is based on incease of the rate constants for Ca^{2+} creation ("k57") from 2.5×10^{-8} M/s to 6.6×10^{-6} M/s every 4th second for a period of 2 seconds. This scheme is reproduced in the RB model by translating the concentration-based rate constants to stochastic rate constant with Equation 2.9. The variable n is set to 0 as the reaction is zero-order. The basal stochastic rate constant Ca^{2+} creation is 15 /s and is increased to 3947 /s for spike train induction. CODE 2.6 shows specification of Ca^{2+} spike train in the RB model.

Unlike Ca^{2+}, cAMP is not present in the system at the beginning of the simulation. It is introduced by addition of 4000 molecules after 200 seconds of simulation. It is encoded in the RB model as in CODE 2.5.

CODE 2.5: Specification of a cAMP pulse

```
1 %mod: [T]=200 do $ADD 4000 cAMP(a, c~on)
```

CODE 2.6: Specification of ten Ca^{2+} spikes

```
1 %mod: [T]>250 do $UPDATE 'k57' 3974
2 %mod: [T]>252 do $UPDATE 'k57' 15
3 ...
```

```
%mod: [T]>286 do $UPDATE 'k57' 3974
%mod: [T]>288 do $UPDATE 'k57' 15
```

2.3.1.5 Simulation settings

Model simulations were executed on Linux Operating System in `bash` command-line with KaSim software version 3.5-250915 with the following command specifying input files and options:

```
#!/bin/bash
KaSim  -i init.ka -i rates.ka -i rules.ka -i stimuli.ka -i
   observables.ka  -t 700 -p 1400 -o ./results.out
```

Five input files are provided followed by the `-i` indicator. These files contain components of model specification: agent signature and their inital abundances (`init.ka`), Variables representing rate constant names with assigned values (`rates.ka`), rules (`rules.ka`), stimuli (`stimuli.ka`), and observables (`observables.ka`). Three remainig option indicators, `-t, -p, -o`, set the time of simulation in seconds (`-t`), the number of data points recorded per observable over the indicated time of simulation (`-p`) and the output file (`-o`). The simulation is set to last 700 seconds with two data points recorded every second resulting with 1400 points reported per observable.

2.3.2 Approach to comparison of models

The results of simulation of a dynamic model are represented in a form of variations of observable quantities recorded per time point called as time-series or time courses. To directly and systematically compare two dynamic model results, their time courses of corresponding observables could be directly superimposed for visual evaluation. However, before it could be performed on the models in question, there are three major aspects to be determined in order to make models become directly comparable in such manner. Firstly, we have to identify analogous observables in two models to compare against each other. Secondly, models are simulated in different schemes, that is deterministic and stochastic. The most preferable condition would be to compare both timeseries simulated in the same scheme. Therefore, either the RB model has to be deterministically solved or the ODE model simulated stochastically. Lastly, models can be modified to illustrate various biological conditions of the same molecular reaction system. By comparing models in variable conditions, we

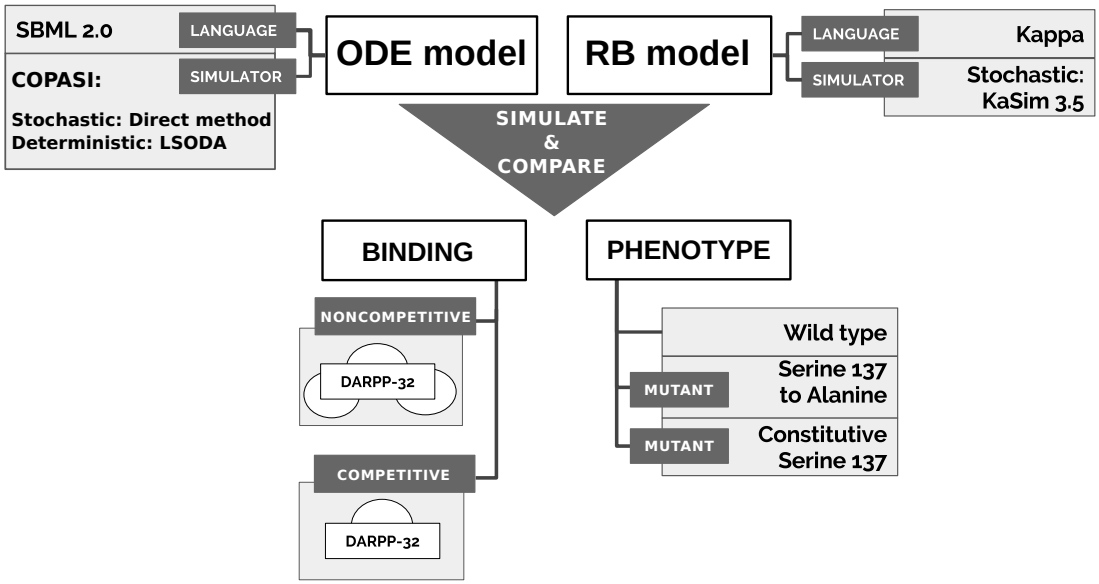


FIGURE 2.6: Approach to comparisons of time-series. Both models are run in a stochastic scheme in different conditions affecting the binding site availability of DARPP-32 and site-mutations of Ser137.

could compare models on broader levels and obtain better understanding of exposed differences.

All these basic requirements are met as outlined on FIGURE 2.6. Both models are simulated in the stochastic scheme but in different simulation environments. The RB model is simulated with KaSim, which is its native simulation environment. Whereas the original model with COPASI, which is a common simulation environment for models defined in the SBML format [112]. Both models are simulated in three different settings reflecting three experimental phenotypes. The base-line setting is called a wild-type and two others represent site-directed mutations applied in the original study of Fernandez et al. [177] (see *Section 2.2.3*). The RB model is additionally tested with two different binding schemes, further called as noncompetitive and competitive binding. In the noncompetitive binding all interactors of DARPP-32 can bind simultaneously to three different sites. The competitive binding assumes that only one interactor can bind at the same time as there is only one binding site. The latter scheme reflects the assumption of the ODE model and therefore, is treated as a baseline scheme of the RB model.

Each of these aspects is described in greater detail in the following

sections.

2.3.2.1 Variable conditions of model

Modifications of model's basal conditions that mimic experimental perturbations is an important feature of a strong modelling framework. Therefore, to extend the spectrum of aspects compared between RB and ODE modelling frameworks, models were subjected to variable types of modifications. The first type is based on modification of rate constants. It is seemingly the easiest approach applied in ODE-based models. This type of modification was used to induce site-directed mutations in the Fernandez model. It is important to establish if this type of modification applied to the ODE model will give similar results when performed on the RB one. Therefore, a similar modification of appropriate rate constants is applied in the RB to emulate site-directed mutations.

The other type of modification applies only to the RB model as it is specific to the RB model notation. It is based on increase of the number of agent's binding sites. This type of modification was not applied in the ODE model as it would require to extend the model with additional molecular species representing molecular complexes that are formed when the number of binding sites is larger than one. As discussed in *Section 1.1* of the introductory chapter and will be discussed in more detail in the coming section (*Section 2.3.2.1*), it is particularly problematic to enumerate and encode the resulting number of molecular species in the ODE-based framework. In this case, the RB model is not compared to the ODE one but rather compared to the RB model that conforms to the ODE model assumption of a single binding site of DARPP-32.

Site-directed mutations As described in *Section 2.2.3*, the Fernandez model was tested with two types of site-directed mutation of Ser137: Serine to Alanine mutation of DARPP-32 at Ser137 (Ser137Ala) and constitutive Ser137 (constSer137). Mutations are designed to induce oppose effects on Ser137. The first one inhibits the site phosphorylation, the second one, leads to its sustained phosphorylation. In both cases, the modification of the model was based on inactivation of particular set of reactions by turning to zero their constant rates.

Reactions suppressed in Ser137Ala mutation are phosphorylation reactions of DARPP-32 at Ser137 by CK1. To introduce this mutation in the

RB model, the catalytic constant of the rule that define phosphorylation of DARPP-32 at Ser137 by CK1 was set to zero ("kcat2"). In the ODE model, it was performed by setting "kcat2", "kcat5", "kcat7" and "kcat14" catalytic constants to zero. These four constants parametrise four phosphorylation reactions of DARPP-32 at Ser137 by CK1. Each reaction has the same reactant, which is a complex composed of DARPP-32 bound to CK1, with the exception that DARPP-32 is in four variations of phosphorylation states at two other phosphorylation sites. These four reactions are represented by one rule in RB model, and therefore, a change of a single constant induced the Ser137Ala mutation.

A similar procedure was performed to induce the constSer137 mutation. In the ODE model suppressed reactions represent dephosphorylation reactions of DARPP-32 at Ser137 by PP2C. The induction of mutation was based on setting "kcat13", "kcat20", "kcat24" and "kcat28" catalytic constants to zero. These four rate constants parameterise four reaction instances of PP2C that dephosphorylate DARPP-32 at Ser137. Similarly to the first mutation, this set of four reactions is defined with a single rule in the RB model. The constSer137 mutation in the RB models was performed by setting to zero a single catalytic parameter, "kcat13".

Variation of the number of agent's binding sites Unlike in ODE model specification, the syntax of rules with the agent signature denoting binding sites and states, brings the protein interface to the attention of a modeller. The Fernandez et al. [177] study does not discuss if DARPP-32 binding partners bind to the same or different active sites. However, based on the ODE model specification, we can say that only one kinase or phosphatase can bind DARPP-32 at a time, as there is no molecular species representing complexes of more than one binding partner (Figure 2.7A). As DARPP-32 is an intrinsically disordered protein with nothing known about its 3D structure, the exact binding interface of DARPP-32 to its partners has not been reported [9, 216–218].

One could ask how a different binding scenarios can change dynamics of the DARPP-32 network. For instance, we could assume a less competitive binding scenario, where kinases and phosphates concurrently bind to their respective phosphorylation sites (Figure 2.7B). To define an ODE model for the DARPP-32 interaction network where all three phosphorylation sites bound at

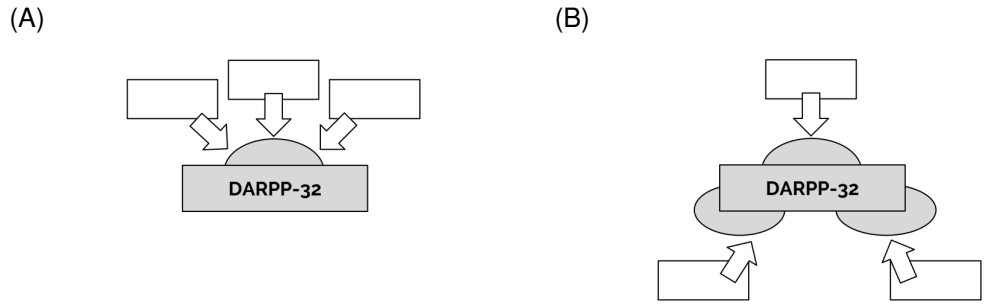


FIGURE 2.7: Two scenarios of binding schemes depicting implications of change in the number of binding sites. In the non-competitive case (A), phosphorylation sites can bind its respective kinases and phosphatases. In the other scenario (B), there is only one site that is able to bind. This induces competition between interactors.

the same time to one of the two possible partners would require addition of new equations and updating the existing ones. There are 42 equations representing molecular species with DARPP-32. The new equations would have to represent additional molecular species of all possible complexes composed of DARPP-32 and three of six other proteins including variations of three phosphorylation sites. The fact that it is difficult to enumerate all these species and that it is unknown if such complexes appear in this interaction network might have decided on the competitive binding assumption in the ODE model.

Contrary to the ODE model specification, a definition of such binding scenario in the RB notation requires exactly the same number of rules, provided that concurrently bound interactors do not influence each other. The major aspect that differentiates definitions of two binding scenarios in the RB syntax is the DARPP-32 agent signature in rules where DARPP-32 is involved. CODE 2.7 shows the complete signature of agent DARPP-32 definition that reflect the original model assumption of a single binding site. Only "s" site can bind, the rest of sites can only alter their internal states between "u" and "p". The equivalent complete signature that represents the alternative scenario of three-binding partners is shown in CODE 2.8. In these case each site with internal state has also binding ability. The first scenario, where DARPP-32 has one binding site, represents a competitive binding and the second, where DARPP-32 has three biding sites, a less-competitive binding of enzymes to the substrate.

CODE 2.7: One-binding site DARPP-32 agent definition.

```
1 %agent: D(s, thr34~u~p, ser137~u~p, thr75~u~p)
```

Code 2.8: Three-binding site DARPP-32 agent definition.

```
%agent: D(thr34~u~p, ser137~u~p, thr75~u~p)
```

We can visualise these differences in the agent binding availability with contact maps (defined in *Section 1.5.1.6* of introductory chapter). Figure 2.8A represents the contact map of DARPP-32 as defined in Code 2.7 and Figure 2.8B as in Code 2.8.

As it is easy with RB modelling to define different versions of binding availability of DARPP-32, the RB model is simulated in both scenarios and corresponding time courses are compared. The following section will present the methodology for direct comparison of model dynamics.

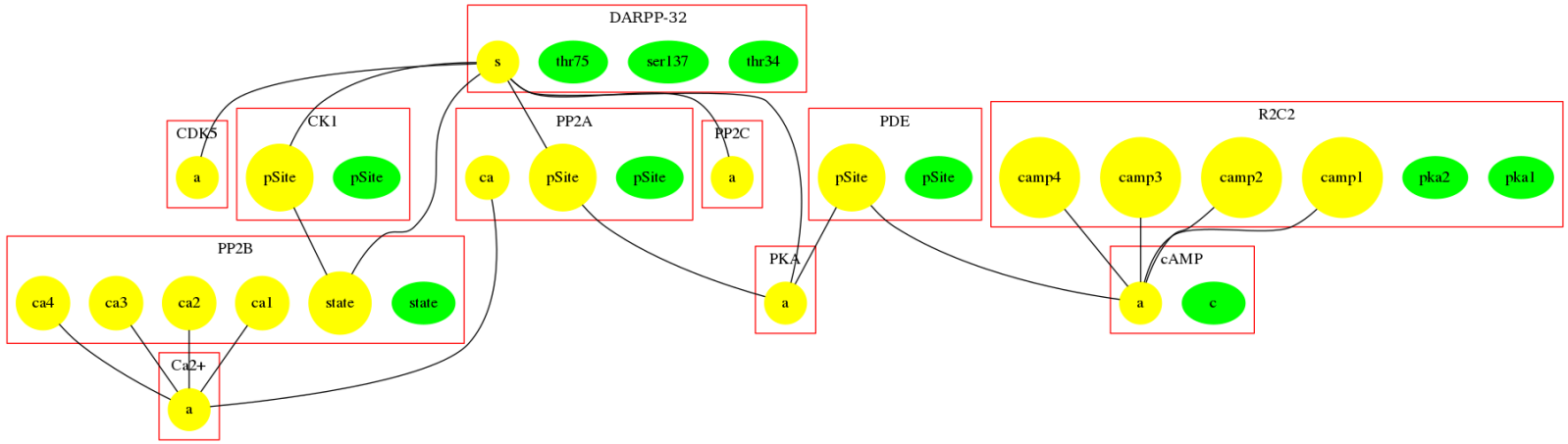
2.3.2.2 Simulating deterministic model with stochastic method

Given the time the model by Fernandez et al. [177] was build in, it is a rare modelling study with a particular stress on reproducibility. As mentioned before, there are in fact two models: "model A" and "model B". The latter being an extension of the former. Both models are provided with the publication and encoded in two formats: SBML (version 2.0) and E-Cell (version 3) [219]. Both are also published in BioModels database [184][2]. They are also available in multiple formats as they passed the manual curation procedure and became automatically translated to other formats such as a molecular pathway standard format BioPAX, and formats readable in other simulation environments like Octave, SciLab, XPP and VCML. In spite of a great selection of available formats, it was unclear whether the model could be run with the stochastic simulation as it was intended and designed to be solved deterministically. Particularly if we consider that 11 years passed since the original publication and a lifespan of scientific software tends to be relatively short.

First attempts of running simulations were primary concentrated on two model formats provided with the publication as directly indicated by the authors. The model was executed in E-Cell version 3 simulation environment. Unfortunately, rerunning the model in this format poses some difficulties. Firstly, E-Cell version 3 requires to specify a simulation file that determines details of the simulation, which is not provided with the publication. It involves first to acquire some experience with the simulation language, which for the 3rd version is poorly documented. The other approach would be to run the

[2]Published under identifiers: BIOMD0000000152, BIOMD0000000153

(A)



(B)

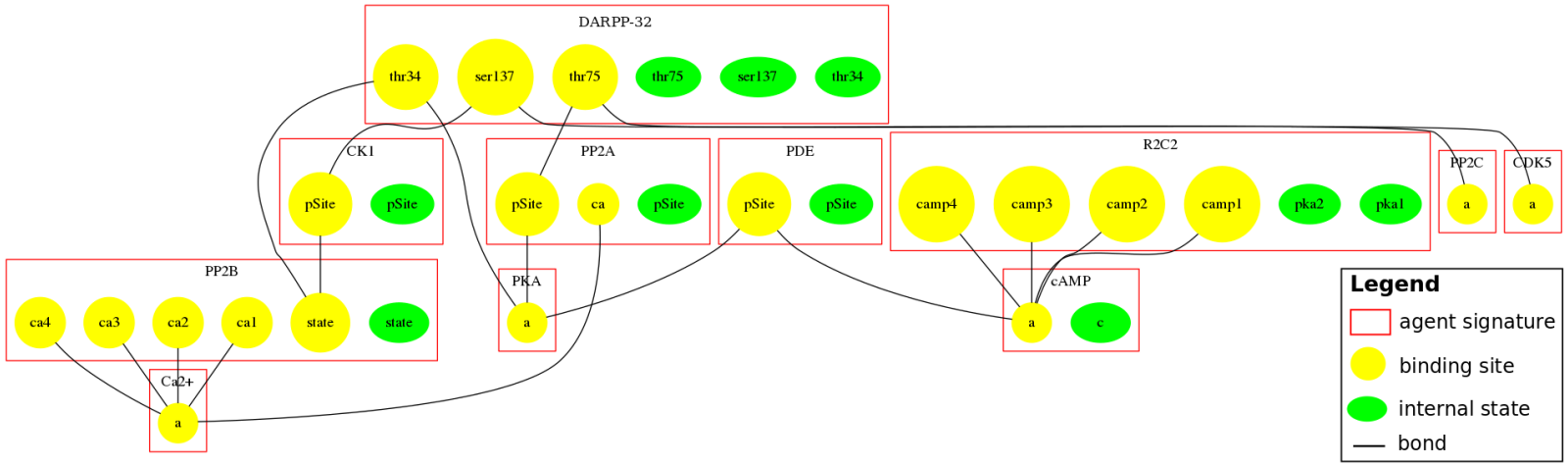


FIGURE 2.8: Contact maps of two RB models for two different binding schemes of DARPP-32 with (A) one binding site and (B) three binding sites. A contact map shows all agents and their signatures composed of binding sites and internal states with connecting lines showing all possible bonds between them.

model with the latest version of the simulator, supported with a more detailed manual. However, the 3rd version is not forward compatible with the 4th version as it does not accept a model encoded in the native format of the 3rd version, with the "`.eml`" file extension, that the model is encoded in.

However, by including the model in the SBML format that is independent of any specific simulation tool or method, the authors have opened a number of alternative routes to simulate the model. Therefore, due to the above mentioned difficulties with the native simulator, SBML was favoured as a more flexible and versatile format. This format allowed testing of multiple stochastic simulation tools designed to load models in the SBML format, among which were COmplex PAthway SImulator (COPASI) [112], Python packages like libRoadRunner [220] and StochPy [221]. With the exception of COPASI in version 4.20, all tested stochastic simulators failed to simulate the model. The most frequently occurring issue was related to the fact that the model includes events executed during simulation. These are encoded in SBML format with the "`Events`" syntax component that triggers modification of an indicated model variable at a certain time point of the simulation. In the Fernandez et al. [177] model, it is used to introduce the stimuli, the cAMP pulse and Ca^{2+} spikes. The same issue was encountered with previous versions of COPASI, which are incapable to run through the Ca^{2+} spiking events. This failure was reported and the next version fully executed the model simulation.

To conclude, the final time-series of the original model were obtained using the SBML model file with COPASI in the version 4.20. COPASI allowed running of the same model with the deterministic solver (LSODA) and stochastic simulator. The stochastic simulator is an implementation of *direct method* introduced by Gillespie [119]. The units of abundances of molecular species were set to be reported in copy numbers, in agreement with the units of observables reported by the KaSim simulator.

2.3.2.3 Selecting and pairing observables

The complete list of molecular species abundances defined to be tracked in the ODE model is 75[3]. These molecular species are composed of variations in binding partners and internal states of 11 molecules. To obtain more general representation of the species of interest and present them in the original pub-

[3]List of molecular species (physical entities) of the Fernandez et al. [177] model on the BioModels website: https://www.ebi.ac.uk/biomodels-main/BIOMD0000000153

lication, the time courses of appropriate species were selected and summed up to represent a single aggregated variable. For instance, "D34" is an aggregated variable that contains counts of all molecular species of DARPP-32 phosphorylated at Thr34 regardless the states of other phosphorylation sites and presence of binding partners. As the ODE model output results indicate, there are 22 molecular species that match the definition of "D34", which are identified by presence of "34" in molecular species names. The concept of aggregated variables corresponds to *observables* in RB modelling and therefore, the term *observable* will be used here also for the ODE aggregated variables. As described in *Section 1.5.1.4* of the introductory chapter, variables traced over the simulation of RB model are defined by a modeller as patterns with a desired level of pattern completeness. For instance, to obtain a similarly defined variable to "D34" from the RB model simulation, an observable is specified with a partial pattern of DARPP-32 agent's interface (Code 2.9) that will represent a sum of trajectories of molecular species that are phosphorylated at the Thr34 site.

Code 2.9: D34 observable definition

```
1 %obs: 'D34' D(thr34~p)
```

In the original publication, there are seven observables that time courses are plotted. Among them, there are three major observables denoting three phosphorylated sites of DARPP-32. They are named as D34, D75 and D137, so that all phosphorylation at the sites could be followed regardless of the states of the other sites. It also means that these observables encompass overlapping sets of molecular species. For example, the D34 observable counts molecular species that are also phosphorylated on Thr75 and Ser137, bound or free. For the same reason, overlapping sets of molecular species compose the other plotted observables, unphosphorylated DARPP-32, PKA, Ca^{2+} and cAMP.

This list of the seven observables was extended to compare dynamics of other molecules defined in the models. The choice of observables was guided by an aim to capture impact of interactions on molecules. It can be generally reduced to the following principles:

- if an agent has internal states, the activated state is set as its observable form, e.g. "`CK1(pSite~u)`"

- if an agent is created and degraded over the simulation, its observable is set to its least specific form, e.g. "PKA()"
- if an agent is not created and degraded during the simulation and thereby its level remains constant throughout the simulation, and has no internal states, its observable is set to its bound form, e.g. "CDK5!_"

To match observables of the RB model to the observables of the ODE model, a list of partial strings was selected to group and sum species trajectories of the ODE model output into a list of aggregated variables. To verify this approach, obtained relevant observables of ODE model with this method were visually compared with the seven observables plotted in the original publication.

TABLE 2.1 presents a full list of names for RB and ODE observables that trajectories are compared, accompanied by their descriptions.

2.4 Results

This section presents the comparison results of the ODE model to the RB one. The models are compared with respect to two major aspects. The first one is a model notation. It is analysed by dividing and comparing sizes of model components. The expected result of this comparison is that the reaction set underlying the ODE model will be represented with fewer rules. This expectation is dictated by the main characteristics of RB modelling, that a single rule can represent multiple reactions by expressing pattern that matches multiple reactions.

The second aspect is based on analysis of model dynamics, where alignment of equivalent time courses obtained by simulating the models is analysed. The comparison of time courses is performed between three variants of each model. The first variant is a base-line condition (wild-type), and two other variants are earlier mentioned site-directed mutations, that act in an opposite manner on the phosphorylation of Ser137 by either blocking it (Ser137Ala) or indefinitely activating (constSer137). Comparing models in these variable conditions can elucidate whether the RB model can be modified in the same manner as the ODE one, that render similar pattern of dynamics. This can potentially augment understanding of differences between the models. Lastly, to leverage the advantage of RB language regarding ease of modifying agents'

RB	ODE	Definition
cAMP*	cAMP	cAMP binding unspecified
free_Ca*	free_Ca	Ca^{2+} unbound
all_Ca*	all_Ca	Ca^{2+} binding unspecified
PKA*	PKA	PKA binding unspecified
CDK5_*	_CDK5	CDK5 bound
CK1u*	CK1u	CK1 unphosphorylated, binding unspecified
PP2Ap*	PP2Ap	PP2A phosphorylated, all bindings unspecified
PP2ACa*	PP2ACa	PP2A bound to Ca^{2+}, phosphorylation and other bindings unspecified
PP2C_*	_PP2C	PP2C bound
PP2Bactive*	PP2Bactive	PP2B active, binding unspecified
PDEp*	PDEp	PDE phosphorylated, binding unspecified
D*	D	DARPP-32 unphosphorylated at all sites, binding unspecified
D34*	D34	DARPP-32 phoshophorylated at Thr34 with unspecified binding, other sites' internal states and binding unspecified
D75*	D75	DARPP-32 phoshophorylated at Thr75 with unspecified binding, other sites' internal states and binding unspecified
D137*	D137	DARPP-32 phoshophorylated at Ser137 with unspecified binding, other sites' internal states and binding unspecified

TABLE 2.1: RB observable names and corresponding ODE observable names with definitions. To obtain the ODE observables, the time-series of appropriate multiple molecular species are summed, based on their names.

ODE model			RB model
Model Component	Total counts	Total counts	Model Component
Reaction instances	152	132	Reaction rules
Concentration-based rate constants	152	62	Stochastic rate constants
Initial concentrations	75	8	Initial copy numbers
Molecular species	75	91/137	Molecular species
Stimuli events	21	21	Stimuli events

TABLE 2.2: Model specification can be divided into components. The total counts of elements in each component is shown for the ODE and the RB model.

binding interfaces, and to reflect on alternative hypothesis of binding abilities of DARPP-32, two variants of the RB model are compared. The first variant corresponds to the assumption implemented in the ODE model. This variant represents DARPP-32 as an agent with a single binding site. The second variant implements a hypothesis where DARPP-32 have three independent binding sites. Results obtained in this section are summarised and further discussed in a separate section.

2.4.1 Comparison of model specification

A model specification can be considered as a first layer of comparison that demonstrates differences between the ODE and the RB models. The specification of models can be divided into components that are typically present in most dynamic molecular models, such as molecular species that trajectories are monitored over the simulation, initial concentrations of these molecular species, and rate constants defining how fast reactions occur. TABLE 2.2 compares the total counts of these components in each of the two models. The ODE model is defined based on 152 elementary irreversible reactions that combined with the same number of rate constants form rate laws. These rate laws are used to define 75 equations, each determining trajectory of one molecular species. All 75 molecular species require initial concentrations to be stated in the model specification. Eight of these species have non-zero initial concentrations, what means that only these eight exist at the beginning of the simulation. This fact is reflected by the total count of agents that initial copy numbers is stated in the RB model notation. The model encoded in the RB language encompasses 152

reactions with 131 rules. As 37 rules are reversible, their definition comprises a single line, that further reduces the model to 94 rules. Each of 132 rules is parameterised by one of 62 unique rate constants. The number of unique rate constants is defined by the outcome of the analysis and translation of reactions instances into rules. The number of unique rate constants is lower than the total number of rate constants used to parametrise the ODE model (152). Recalling criteria of condensing reactions into rules described in *Section 2.3.1.2*, multiple reaction instances can be replaced with one rule, if these reactions are of the same type, occur between the same pair of reactants, and are parametrised with the same values of rate constants. What follows, multiple rate constants can be replaced with a single one. The final rule set is more than two fold larger than the claimed here unique number of rate constants (compare 132 rules to 62 rate constants). This means that more than one rule is parametrised by the same rate constant.

As opposed to the ODE model, the number of molecular species in the RB model is not given in the model specification. This number can be obtained only as an approximation obtained with analytical estimation or with use of snapshots to sample the state of molecular mixture over the simulation (see *Section 1.5.1.5* for the snapshot definition). To obtain the maximal number of molecular species presented in Table 2.2, multiple snapshots are repeatedly taken from the start to the end of the simulation with a regular interval, every 10000th simulation event ("[E]") (see *Section 1.5.1.5* for the event definition). The instruction defining a procedure of snapshot sampling was encoded in the model definition as in Code 2.10.

Code 2.10: Definition of snapshot sampling

```
%mod: repeat [T]>0 && ([E] [mod] 10000)=0
      do     $SNAPSHOT
      until  [T]=700
```

This sampling resulted with 9322 snapshots that were parsed to obtain unique expressions of molecular species. This procedure has shown that the total number of molecular species is 91 for the competitive RB model, and 137 for the non-competitive one. In both cases, the sum of molecular species is higher than in the ODE model (75).

The total count of stimuli events in the RB model was reproduced as in

the original publication (detailed in *Section 2.3.1.4*). In both models, stimuli events are composed of one cAMP pulse, ten rises and ten drops of the Ca^{2+} influx constant resulting in a total of 21 stimuli events.

In summary, there are two main observations that can be derived from Table 2.2. Firstly, that the number of rules corresponding to reactions is lower. Secondly, that the number of molecular species is much higher. These remarks could have been easily anticipated based on the essential theoretical difference contrasting the ODE and the RB modelling methods. The ODE-based modelling framework requires an explicit encoding of a complete list of reaction instances, molecular states and complexes that can appear in the modelled system. Therefore, existence of each has to be known and defined in the model specification. On the other hand, the RB model notation allows for representation of multiple reaction instances with one reaction pattern that contains only necessary reaction context, thereby reducing the number of rules to encompass the same number of reactions.

This capability of generating reaction instances with reaction patterns is linked with the extended number of possible molecular species in the RB model, compared to the ODE model. As a rule refers to partially defined agents that in the molecular mixture might exist in variable forms, it can be applied in many different contexts and generate molecular species, that are not defined anywhere in the model specification but *emerge* over the simulation.

It can be observed from the model specification that the reaction pattern notation exposes differences between reactions by hiding details that are default but irrelevant in the reaction context. In consequence, the model specification can be represented more clearly than with ODE. Moreover, a rule notation is much closer to the notation of chemical reactions than equations. Therefore, it is potentially much easier to encode a set of chemical reactions with rules than with equations if a modeller is a biochemist.

However, implicitness of details in the rule and observable notation requires from the modeller to be aware of what is hidden and, if not accounted for, might produce unintended effects. For example, in the case of observable specification, let assume an agent that has one site that can be in two states and is also a binding site in the rule representation. If we would like to write an observable that tracks the agent in one of the two internal states, regardless its binding state, then not only the site with the desired state has to be mentioned

	Model component	Reactions	Rules	Unique rate constants
1.	DARPP-32 phosphorylation	84	27	27
2.	CK1 phosphorylation	4	4	4
3.	PDE phosphorylation	4	4	4
4.	PP2A phosphorylation	4	4	4
5.	PP2B activation	4	24	4
6.	PKA activation	12	56	7
7.	cAMP & Ca^{2+} degradation	8	8	8
8.	PP2A activation by Ca^{2+}	32	4	4

TABLE 2.3: The list of reactions in the Fernandez et al. [177] publication was divided into components based on more general molecular processes represented by subsets of reactions, such as phosphorylation or activation. We can examine reaction to rule relation more closely by comparing models by components. The table shows counts of reaction rules to reaction instances and unique rate per model component. It can be noted that reduction in the number of reaction instances due to translation of reactions into the Kappa language occurred only in two model components (1. & 8.) but in two others resulted in extension of the rule number (5. & 6.).

but also it has to be explicitly marked with "?" to denote its bond-indifference.

```
%agent: A(pSite~p~u)      # pSite servers as binding site and has
                          # internal state

%obs: 'Ap' A(pSite~p)     # pSite in state 'p' and free
%obs: 'Ap' A(pSite~p?)    # pSite in state 'p' and unspecified
                          # binding state
```

This can be avoided by creating separately sites with internal states and sites with binding states (CODE 2.4.1).

```
%agent: B(b, pSite~p~u)   # b servers as binding site
                          # pSite has internal state

%obs: 'Bp' B(pSite~p)     # pSite in state 'p' and unspecified
                          # binding state
```

Despite the advantage of rule notation for representation of reaction in-

stances, the number of rules is only slightly lower than the number of reactions (152 to 131). One would expect that with the reaction pattern notation, this number could have been much lower. It appears that if we closely compare models by parts representing more general molecular mechanisms, the translation of reactions into Kappa language rule patterns results in reduction of the reaction number in some components but extension in others. TABLE 2.3 juxtaposes counts of reaction instances with rules and unique rate constants per model component. This particular division of model representing molecular mechanisms was defined by the original publication. We can see that the reduced representation of reactions obtained with rules occurred in only two components, "DARPP-32 phosphorylation" and "PP2A activation by Ca^{2+}". Both parts mainly contain reactions where DARPP-32 is one of the reactants. To represent a reaction between DARPP-32 and a reactant, all combinations of states on DARPP-32 phosphorylation sites has to be named. This results in a relatively high number of reaction instances but not reaction rules.

"PP2A activation by Ca^{2+}" is a supplementary reaction set that was added to the "model A" version of the original ODE model. Reactions of the "PP2A activation by Ca^{2+}" component and "model A" together compose "model B", the version of ODE model compared in this study. The "PP2A activation by Ca^{2+}" reaction set is composed of 32 reactions. In general, these set of reactions introduces an mechanism where PP2A binds to Ca^{2+} that results in a four fold decrease of the constant rate in reactions where PP2A dissociates from D75 ("`koff`"). This makes PP2A more likely to dephosphorylate Thr75. The activity of PP2A in both "model A" and "model B", is enhanced by its phosphorylation. In "model B", it is further amplified by Ca^{2+} binding. All this information is encoded both by manipulation of rates, and the structure of reaction network. In the RB model, this reaction set was represented as additional four rules and four rate constants. To include complexation of PP2A with Ca^{2+}, additional binding site reserved for the ion was introduced in the PP2A agent signature.

In these two cases, where rule representation reduced the number of reactions, the number of unique reaction rates is exactly the same as the number of rules (column: "Unique rate constants" in TABLE 2.3). The number of unique rate constants reflects the number of differences existing between reactions, and therefore appears to set the threshold for the minimum number of rules

that reactions can be rewritten to.

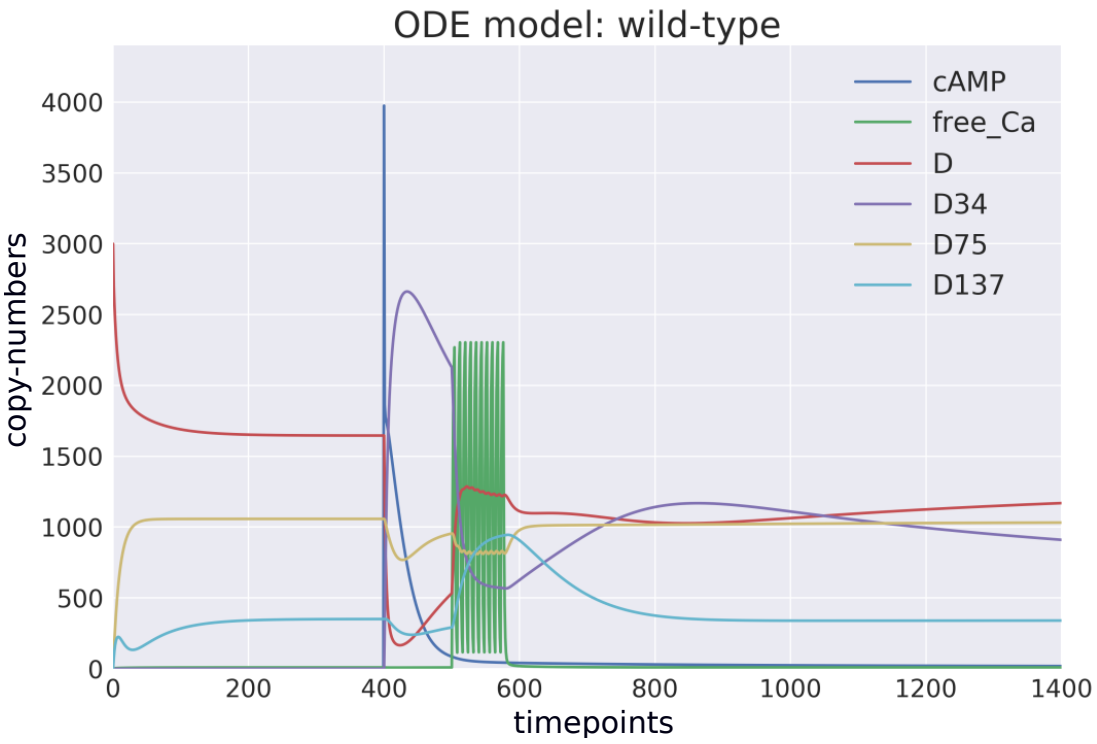
A different situation takes place in the other extreme cases where the number of chemical reactions expanded in rule representation. Here the amount of rate constants is much lower then rules. This happens in the "PKA activation" and "PP2B activation" components. They both have four sites that bind the same molecules, Ca^{2+} and cAMP respectively. This kind of agent structure requires combinatorial binding in the rule notation, described in *Section 2.3.1.2*. It is an example where RB language does not reduce and clarify the model specification, what contradicts the expectation.

2.4.2 Comparison of trajectories

Figures 2.9 shows observable trajectories from simulations of the wild-type ODE model obtained with a deterministic solver (Figure 2.9A) and a stochastic simulator (Figure 2.9B). As described in *Section 2.3.2.3*, trajectories of multiple molecular species of ODE model were summed to correspond to the relevant trajectories presented in the original publication. All stochastic trajectories are reported based on approximately 40 model simulations, where a dark trance is a mean value and shade is a standard deviation from the mean. These trajectories reveal general characteristics of model dynamics. In the first 400 time-steps, the balance between unphosphorylated DARPP-32 ("D") and two phosphorylated forms, "D137" and "D75", settles to a steady state. After 400 time-steps, the cAMP stimulus is applied, what is manifested as a sharp peak. This leads to a similarly sharp phosphorylation of Thr34 ("D34"), and a mirrored drop in unphosphorylated DARPP-32 ("D"). After the next 100 time-steps, Ca^{2+} spiking occurs causing a sudden drop in "D34", recovery of "D", distinctive increase of D137, and slight decrease of "D75". As the stimulus ceases entering the relaxation phase, "D34" levels rebound to reach its second peak at 800th second, reaching the level of "D". Similar behaviour can be clearly seen in both figures (compare Figures 2.9A and Figure 2.9B). However, the standard deviation in the stochastically simulated ODE model reveals a distinctive variation in abundance of the "D34" observable during the relaxation phase.

Figure 2.10 presents the ODE model juxtaposed with the RB model general dynamic traces. Albeit some visible differences, the RB model generally recapitulates dynamics of the ODE model. Furthermore, there is a comparable

(A)



(B)

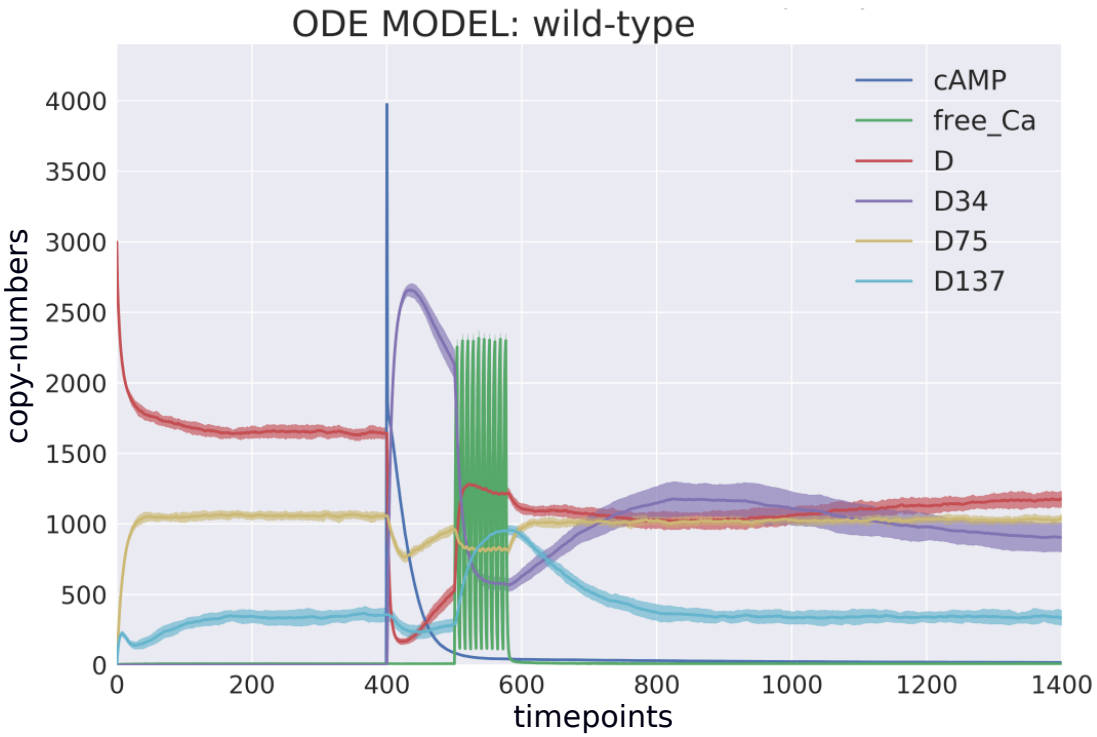
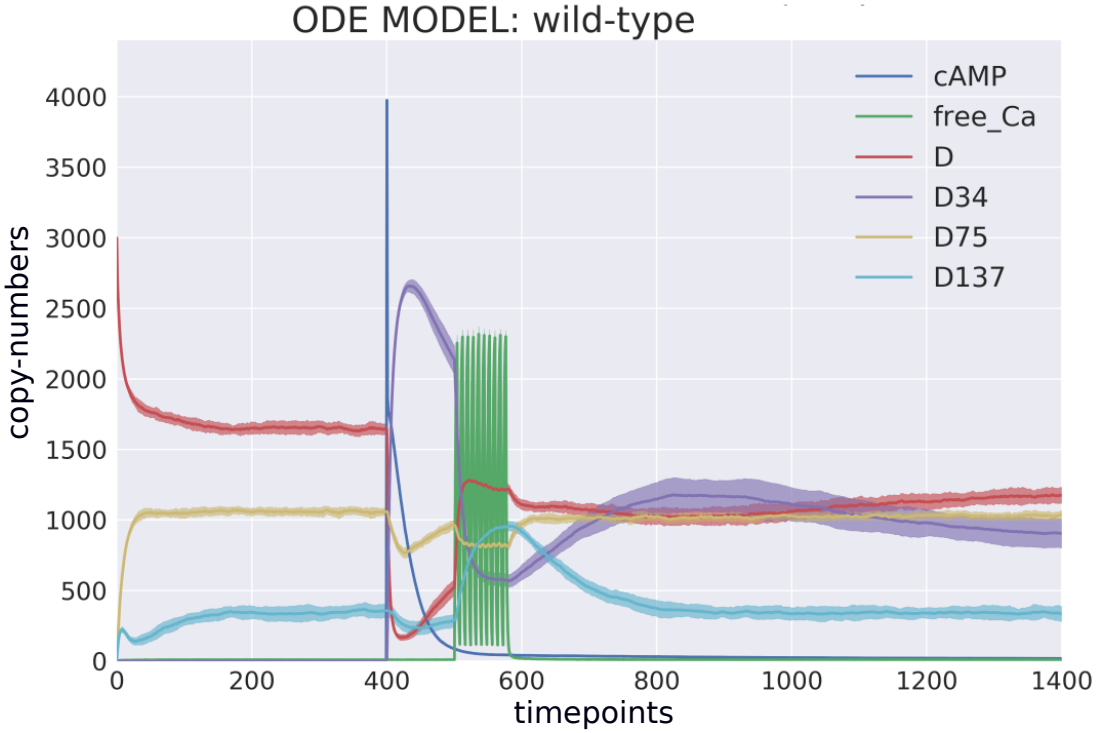


FIGURE 2.9: General dynamics of the ODE model obtained with (A) a deterministic solver, and (B) a stochastic simulation. Trajectories of the stochastic simulation were obtained from calculating mean value (*line*) and standard deviation (*shade*) based on 40 simulations.

(A)



(B)

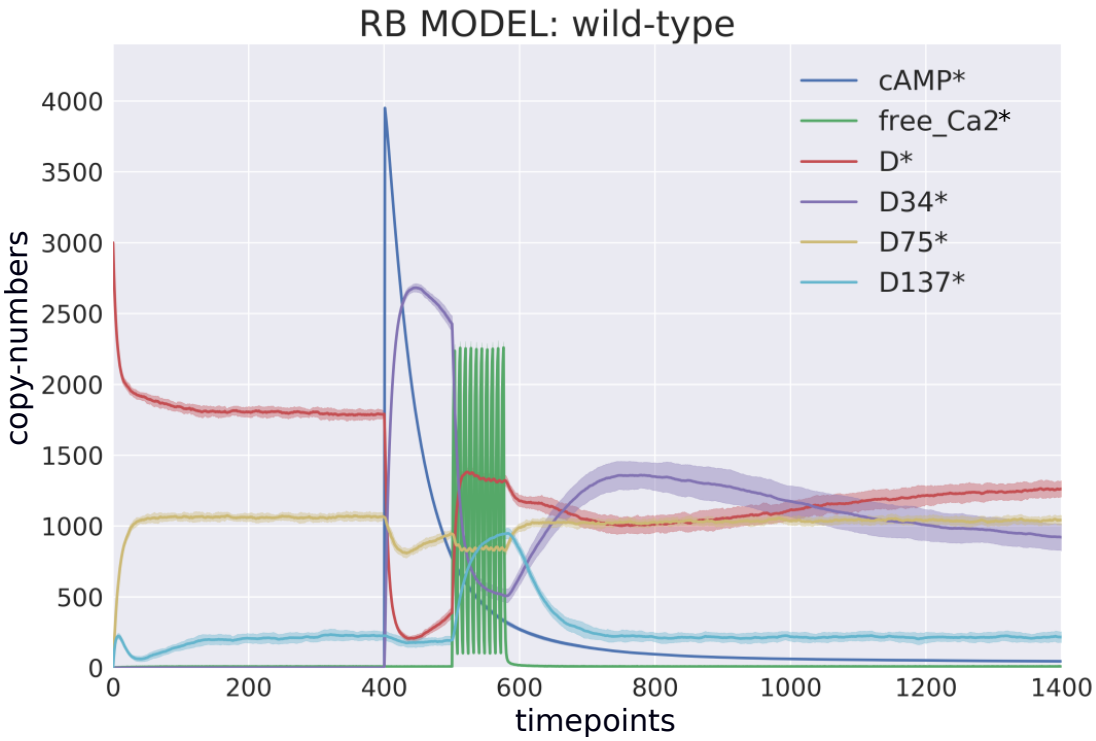


FIGURE 2.10: General dynamics of (A) ODE model and (B) RB model, both in stochastic setting.

variability of "D34" and "D34*" that can be observed in the relaxation phase. "D34" in the ODE model needs 100 more time steps to reach the second peak than its RB counterpart ("D34*"). In the RB model, the peak is slightly higher than in the ODE model. Lastly, the drop of the total number of cAMP molecules in the ODE model ("cAMP") is much sharper than in the RB model.

For a closer look at the model simulations, fifteen traces of observables (defined in Table 2.1) obtained from ODE and RB simulations were paired and superimposed (Figure 2.11). The results confirm the close agreement between models. However, next to the clear matches (e.g. Figure 2.11: B, E, H, N), there are discrepancies between paired curves. Five of these fifteen observables (Figure 2.11: C, F, I, J, O) are examples of the largest discrepancies between models, and are shown separately on Figure 2.12. "all_Ca" and "all_Ca*" observables in Figure 2.12A denote all Ca^{2+} ions present in the system during the simulation. From the beginning of the simulation, a weak Ca^{2+} influx and outflux causes a constant presence of ions in low quantities. At the 500th time point, the Ca^{2+} spiking is induced by a large Ca^{2+} influx. This process is only visible in the RB observable that trajectory elevates above zero and remains a steady state until the Ca^{2+} stimulus occurs at the 500th time point. The "all_Ca" observable trajectory of ODE model remains flat and rises as the Ca^{2+} spiking what resembles rather the observable representing abundances of free Ca^{2+} (Figure 2.11B). The other four observables (Figure 2.12: B, E, H, N) of two models have similar pattern of behaviour though their levels are noticeably lower in the RB model than in the ODE one. All four reflect general characteristics of the "all_Ca*" observable trajectory of the RB model what suggests their dependence on the Ca^{2+} abundance. A fragment of the model reaction diagram in Figure 2.13 demonstrates that these four observables are indeed directly connected in a chain of activation reactions that begins with the Ca^{2+} ions that activate PP2B.

On the ODE observable list, an activated PP2B is named as "PP2Bactive". "PP2Bactive" dephosphorylates CK1, represented by the "CK1u" observable. "CK1u" phosphorylates D137 (the "D137" observable), which is dephosphorylated by PP2C. As the quantity of PP2C is unchanged over the simulation, the observable representing PP2C is encoded as its bound form, named "_PP2C". Due to this chained dependence between reactions, higher abundance of "all_Ca" in the ODE model could explain observed differences between these 4 observ-

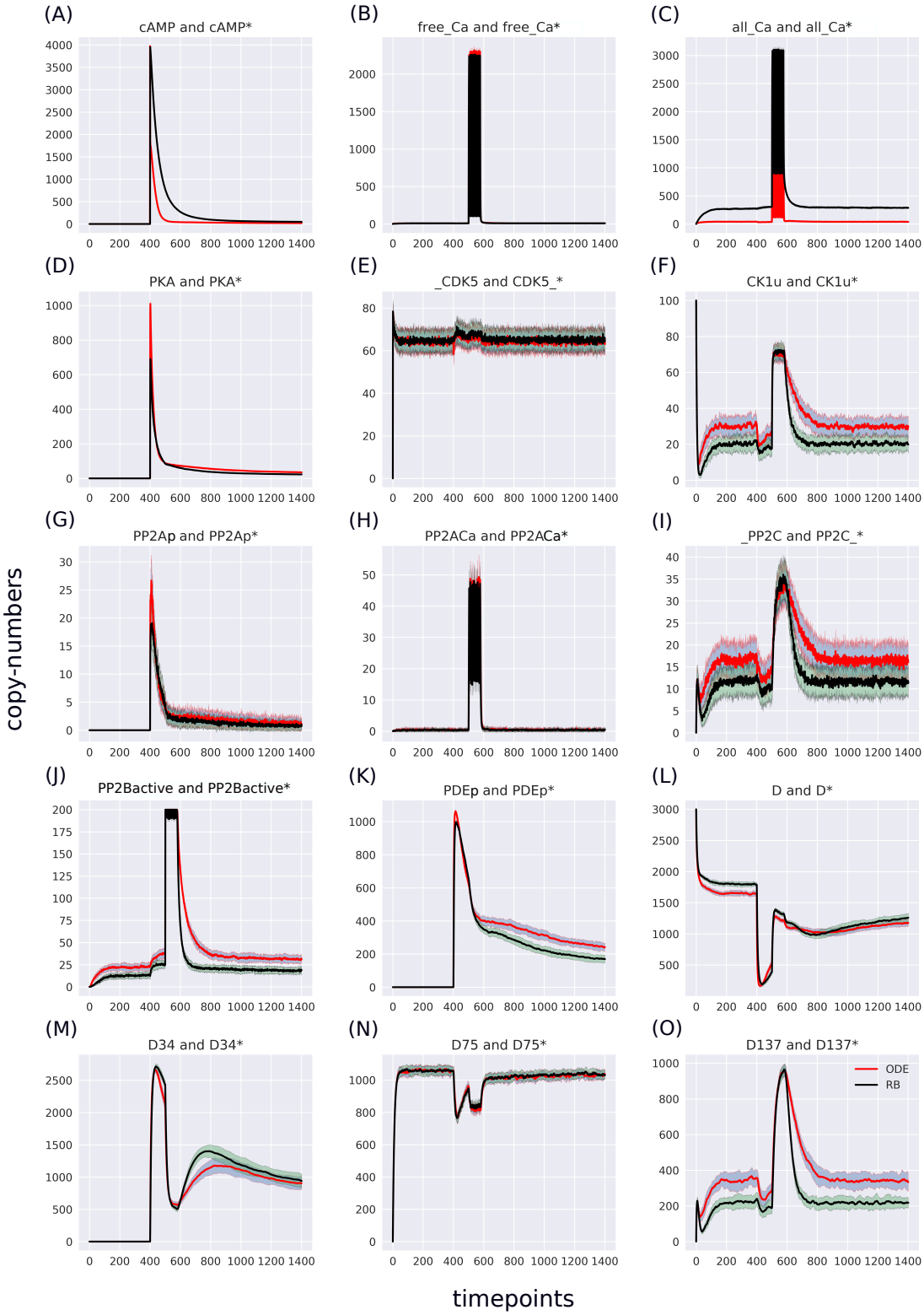


FIGURE 2.11: Superimposed ODE stochastic and RB stochastic time courses in the base-line condition. Note that the scales on y-axis are different to closely visualise traces of the observables. Trace colour: ODE (*red*), RB (*black*).

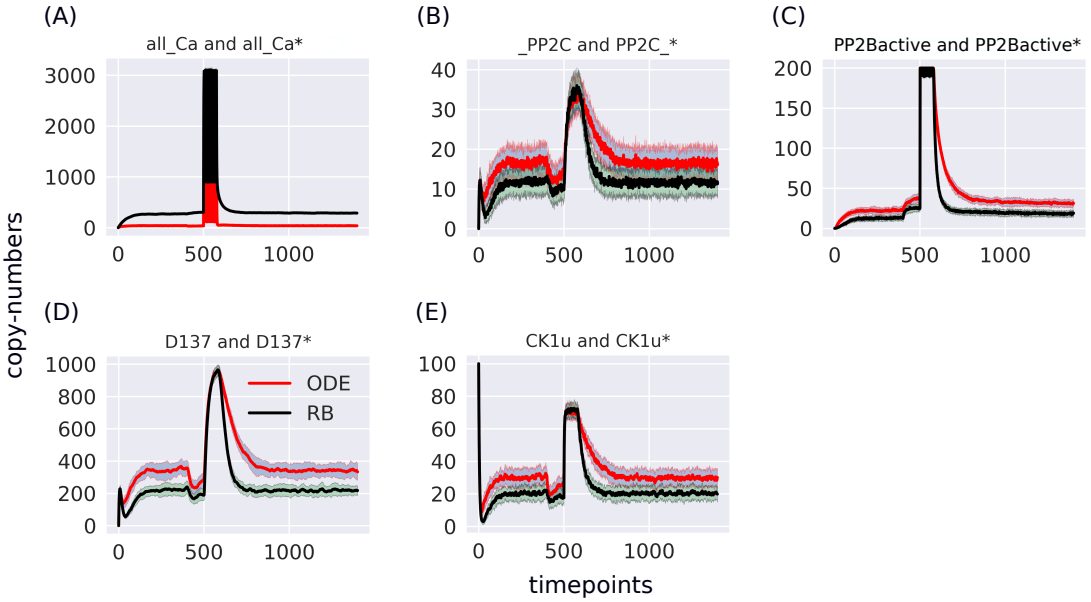


FIGURE 2.12: The largest discrepancies between models can be seen in these observables. The difference between the RB observable "all_Ca2+*" and the ODE observable "all_Ca", denoting all Ca^{2+} ions present during the simulation is further investigated in this section.

ables. However, the abundance of "all_Ca" during steady states in two models are opposite, lower for the ODE observable and higher for the RB one. The next in the chain of responsibility is "PP2Bactive". The higher level of active PP2B is in agreement with the other three observables, what suggests that the observable should be further examined as a potential factor generating these discrepancies between models. Based on the trajectories of the ODE model, we can reason that a stronger activation of PP2B results in a proportionally more copies of the unphosphorylated CK1 and phosphorylated D137. This in turn results in increase of substrate availability for PP2C and therefore, more copy-numbers of its bound form. This effect is inverted in the trajectories derived from the RB model.

It remains unclear why the "all_Ca" observable trajectory produced by the ODE model is much lower than in the RB model at the steady state. Moreover, "PP2Bactive" appears to dictate the higher effect on the other three observables ("D137", "CK1u", "_PP2C"). Therefore, of all five observables, "all_Ca" and "PP2Bactive" and their RB counterparts, seems to be most important in explanation of differences between two model simulation results and therefore, they are closer analysed in further steps.

According to the reaction set underlying both models, the activated PP2B is a complex composed of four Ca^{2+} ions and PP2B. In the ODE model, a name of variable representing an active form of PP2B does not explicitly indicate that it is a complex harbouring Ca^{2+} ions. Therefore, actual copy numbers of all Ca^{2+} ions in the simulated system should also include copies of "PP2Bactive". This would have to be obtained with the analysis of relevant reaction context of the ODE model, not only by summing copy-numbers of molecular species that variable names indicate the presence of Ca^{2+}, as it was done so far.

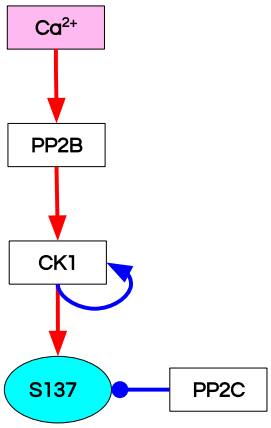


FIGURE 2.13: Reaction diagram connecting observables that exhibits the largest discrepancy between corresponding observables of ODE and RB models. These observables are connected in a chain of activation reactions dependent on each other and triggered by the Ca^{2+} influx.

In the RB model, the composition of each complex species is explicitly represented in the simulation, where obtaining an observable of interest is an automated procedure that sums trajectories matching the observable expression pattern. In the RB simulation, an active form of PP2B is represented as in CODE 2.11.

CODE 2.11: Active form of PP2B

```
1 PP2B(ca1!0, ca2!1, ca3!2, ca4!3, state~a),
2 Ca2+(a!0), Ca2+(a!1), Ca2+(a!2), Ca2+(a!3)
```

With this explicit representation, an observable that is set to track copy numbers of all Ca^{2+} ions during the simulation of the RB model will include ions bound to PP2B. Therefore, the comparison of "all_Ca" to "all_Ca*" observable trajectories (FIGURE 2.12A) is inaccurate due to a difference in molecular species included in these observables. Similar inaccuracy, related to naming observables, explains the divergence between time courses of observables of the total number of cAMP (FIGURE 2.10). Contrary to the RB model, multiple copies of cAMP bound to R2C2 are not included in this time course of the ODE model (compare "cAMP" and "cAMP*").

As a complete list of molecular species in the RB model is not included in the specification, it is also unknown if there are other trajectories of molecular species summed in "all_Ca*". To obtain such approximated list, all molecu-

lar species containing Ca^{2+} in the RB model simulation were isolated from snapshots as defined in Code 2.10. The number of these isolated species is 24, whereas in the ODE model, the number of molecular species composing the "all_Ca" observable is 13. Of these 24 species, 18 correspond to 13 species composing the ODE observable. These 6 molecular species (24 – 18), absent in the "all_Ca" observable, are composed of an active form of PP2B containing four Ca^{2+} ions, either free or bound to phosphorylated CK1, or DARPP-32 in four different combinations of phosphorylation states. These 18 species of RB model that correspond to 13 species of the "all_Ca" observable is higher by 5 because of the difference in number of species that represent a half-active form of PP2B bound to two Ca^{2+} ions. In the ODE model, a half-active PP2B is represented as a single species, with a variable named as "PP2BinactiveCa2". The same species in the RB model exists in six variants (Code 2.12).

Code 2.12: Molecular species of the half-active PP2B in the RB model

```
1 PP2B(ca1  ,ca2!0,ca3!1,ca4  ,state~i),Ca2+(a!1),Ca2+(a!0)
2 PP2B(ca1!0,ca2  ,ca3!1,ca4  ,state~i),Ca2+(a!0),Ca2+(a!1)
3 PP2B(ca1  ,ca2  ,ca3!0,ca4!1,state~i),Ca2+(a!1),Ca2+(a!0)
4 PP2B(ca1!0,ca2  ,ca3  ,ca4!1,state~i),Ca2+(a!0),Ca2+(a!1)
5 PP2B(ca1!0,ca2!1,ca3  ,ca4  ,state~i),Ca2+(a!0),Ca2+(a!1)
6 PP2B(ca1  ,ca2!0,ca3  ,ca4!1,state~i),Ca2+(a!0),Ca2+(a!1)
```

Now that we extracted molecular species composing the "all_Ca*" observable, we can compare exactly the same trajectories of molecular species between the two models by simulating the RB model with a set of new observables, defined precisely as these composing the "all_Ca" observable in the original ODE model (Figure 2.12A). To match this newly defined RB observable of "all_Ca*" to the original model, these 18 species were summed to obtain a single observable (Figure 2.14B). In comparison to the unaltered species composition (Figure 2.14A), the result show that discrepancy between the ODE and RB observable trajectories has diminished. Knowing that there is a higher number of forms representing the half-active PP2B in the RB model, we can try to obtain a closer match between the "all_Ca" observables by superimposing only one of six trajectories of the RB model. Figure 2.14C shows that the match is close to perfect. It demonstrates that the differences between the "all_Ca" observables of the two models can be explained by the difference in the number of

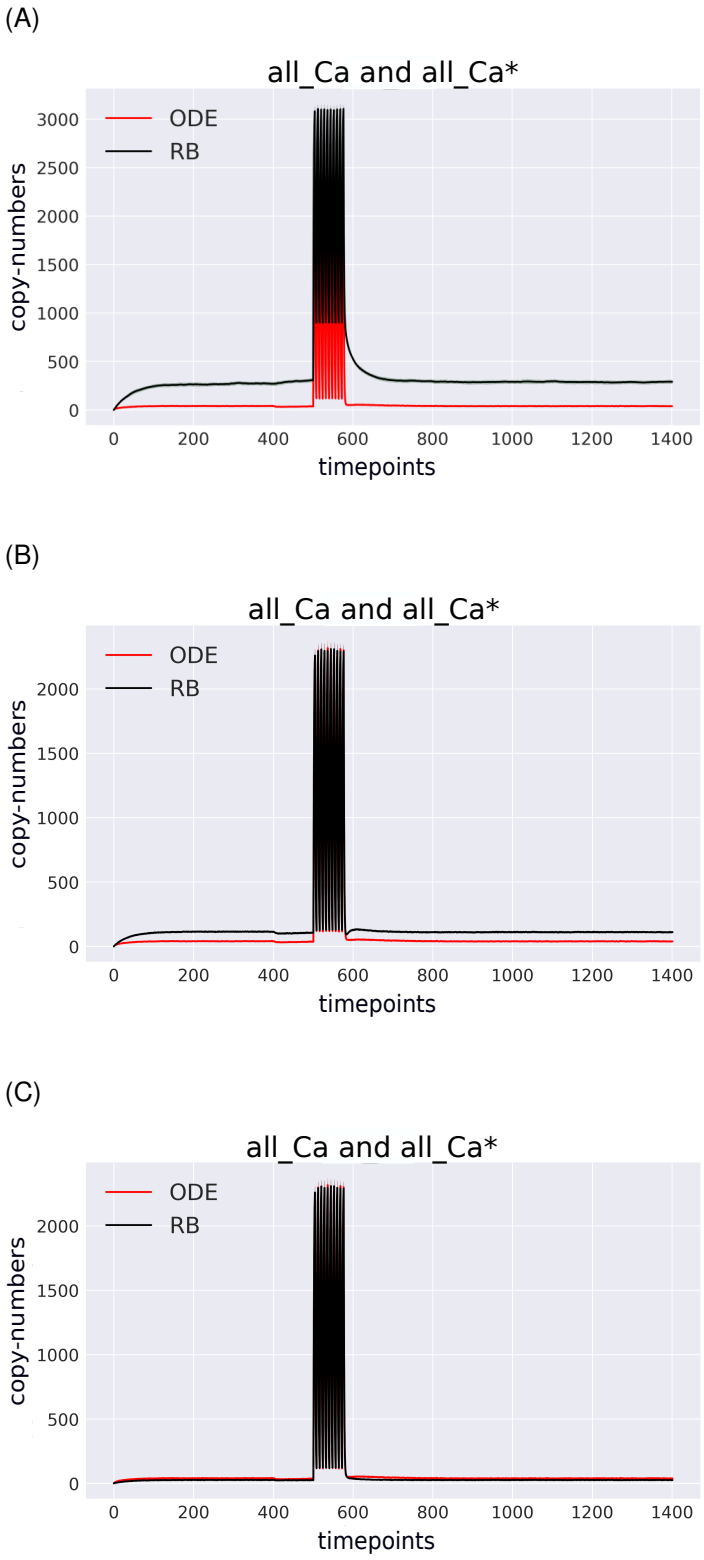


FIGURE 2.14: Comparison of variable compositions of molecular species containing Ca^{2+} ions tracked in the system in both models with (B) unaltered observables; (B) all molecular species containing Ca^{2+} ions selected exactly as indicated by names in the original model and summed to obtain a single trace, where 13 molecular species in the ODE model are represented by 18 species in RB model; (C) 13 molecular species of ODE model matched to 13 of RB model, where only 1 of 6 molecular species of inactive PP2B was selected.

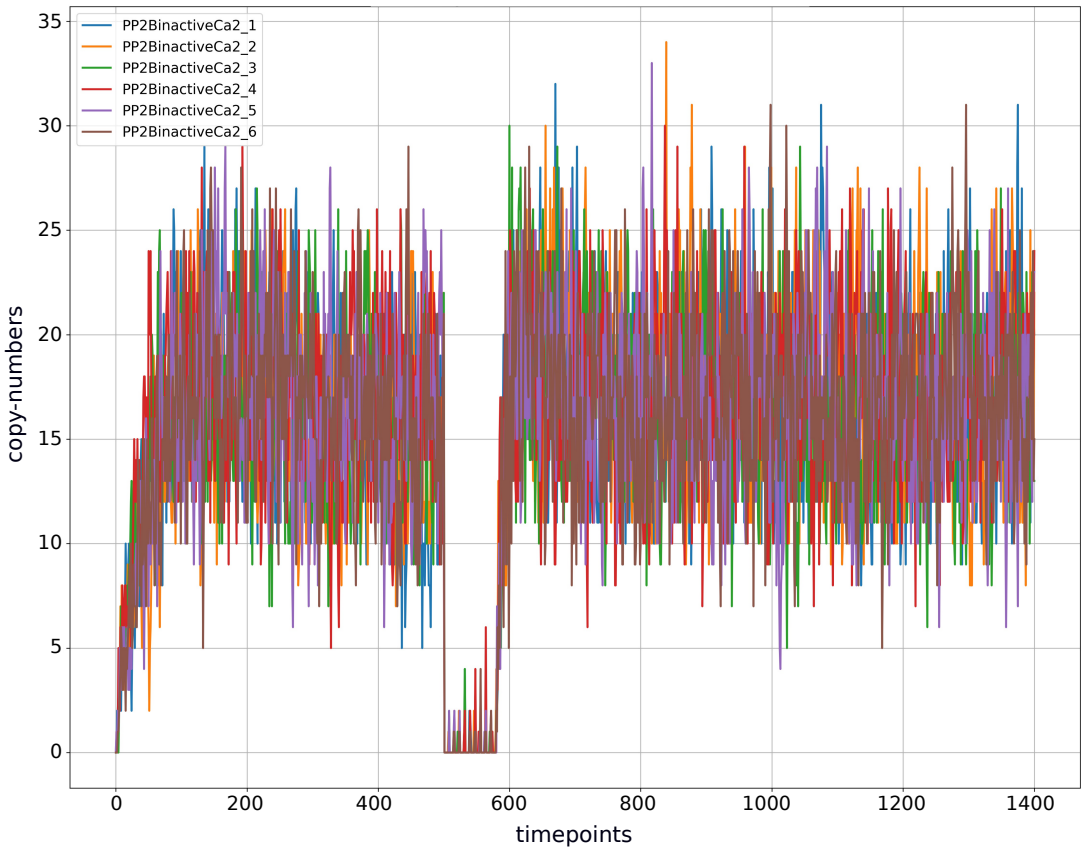


FIGURE 2.15: A half-active PP2B is a complex composed of PP2B and two Ca^{2+} ions. The RB model simulation generates six different molecular species that represent this complex, due to a combinatorial binding of Ca^{2+} ions to four identical sites of PP2B. The plot shows a superimposed trajectories of these six variants of the half-active PP2B. Neither of these six trajectories differentiates itself from others by a pattern of dynamics nor an average level of abundances.

representations of the molecular species. To choose one of the six trajectories, differences between them were examined in FIGURE 2.15. As they all have a similar pattern of dynamics and the same average levels of abundances, the choice of one of these forms is assumed as arbitrary. The distinction between locations of two Ca^{2+} ions on the numbered sites of PP2B, as enforced by encoding of Kappa (CODE 2.12), is irrelevant since all four sites are functionally indistinguishable.

FIGURE 2.16 shows 13 pairs of traces of corresponding molecular species of the two models. The largest discrepancy between trajectories can be observed in "PP2BinactiveCa2" (FIGURE 2.16M) that for the RB time course was

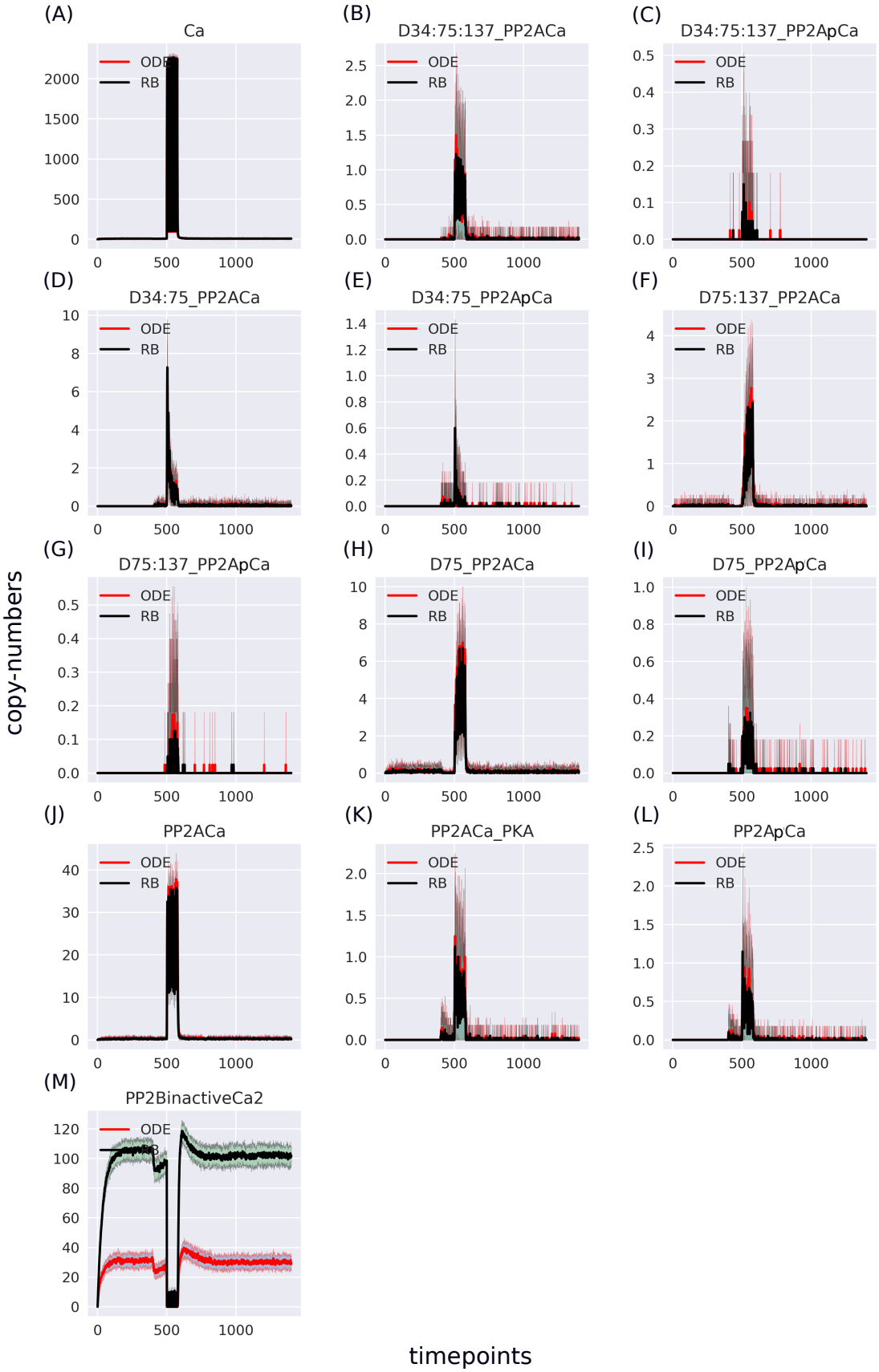


FIGURE 2.16: Separated molecular species containing Ca^{2+}, selected as in the original model. The largest discrepancy lays in the "PP2BinactiveCa2" species, which is the summation result of six entities representing an inactive form of PP2B in the RB model.

obtained by summing six entities representing a half-active PP2B into one. If we again choose one of the six trajectories and plot corresponding molecular species in pairs (FIGURE 2.17M), though diminished, differences between them remain. This time the trajectory of the RB model is lower then the one of the ODE model.

These six forms of the half-active PP2B suggest that a better match between the two models can be achieved by decreasing the constant rate of rules that represent binding of Ca^2+ to free PP2B. However, this could bring the desired effect solely for the half-active PP2B ("PP2BinactiveCa2") but not for dynamics of other observables if their parameters were not adjusted simultaneously. In particular, the fully active PP2B ("PP2Bactive"), of which the "PP2BinactiveCa2" is an intermediate form. With the current parameter values, the fully active PP2B in the RB model is lower than in the ODE model (FIGURE 2.11J). Decreasing the rate constant of its intermediate form would lead to further decease of its copy numbers. It is important to observe that even though there are more representatives of the half-active PP2B in the RB model than in the ODE one, its fully active form has lower levels. It is so despite that a rate constant that produces the fully active PP2B is three-fold higher than for the inactive PP2B.

To further examine this paradox, we can return to the comparison of model specifications (*Section 2.4.1*). The activation of PP2B in the rule representation belongs to most divergent from the reaction representation. If we recall the juxtaposition of reaction to rule counts per model component (TABLE 2.3), the "PP2B activation" part is encoded with 24 rules, 4 reactions and 4 unique rate constants. Of these 24 rules (4 reactions), 12 (2) represent the first step of binding and dissociation of two Ca^{2+} ions from PP2B. The Other 12 (2) represent the second step of binding and dissociation of another two Ca^{2+} ions from PP2B. The activation of PP2B in the rule representation is dissected to a site-specific detail that include all combinations of positioning Ca^{2+} on four sites of PP2B. This suggests that the larger number of rules defining transition from inactive to active PP2B slows down this process in the rule representation.

To support this hypothesis that the number of rules activating PP2B might be the reason for the lower level of "PP2Bactive*" in the RB model, we can review the second example that required much larger number of rules, that is an activation of PKA. FIGURE 2.11D shows that the RB trajectory of the

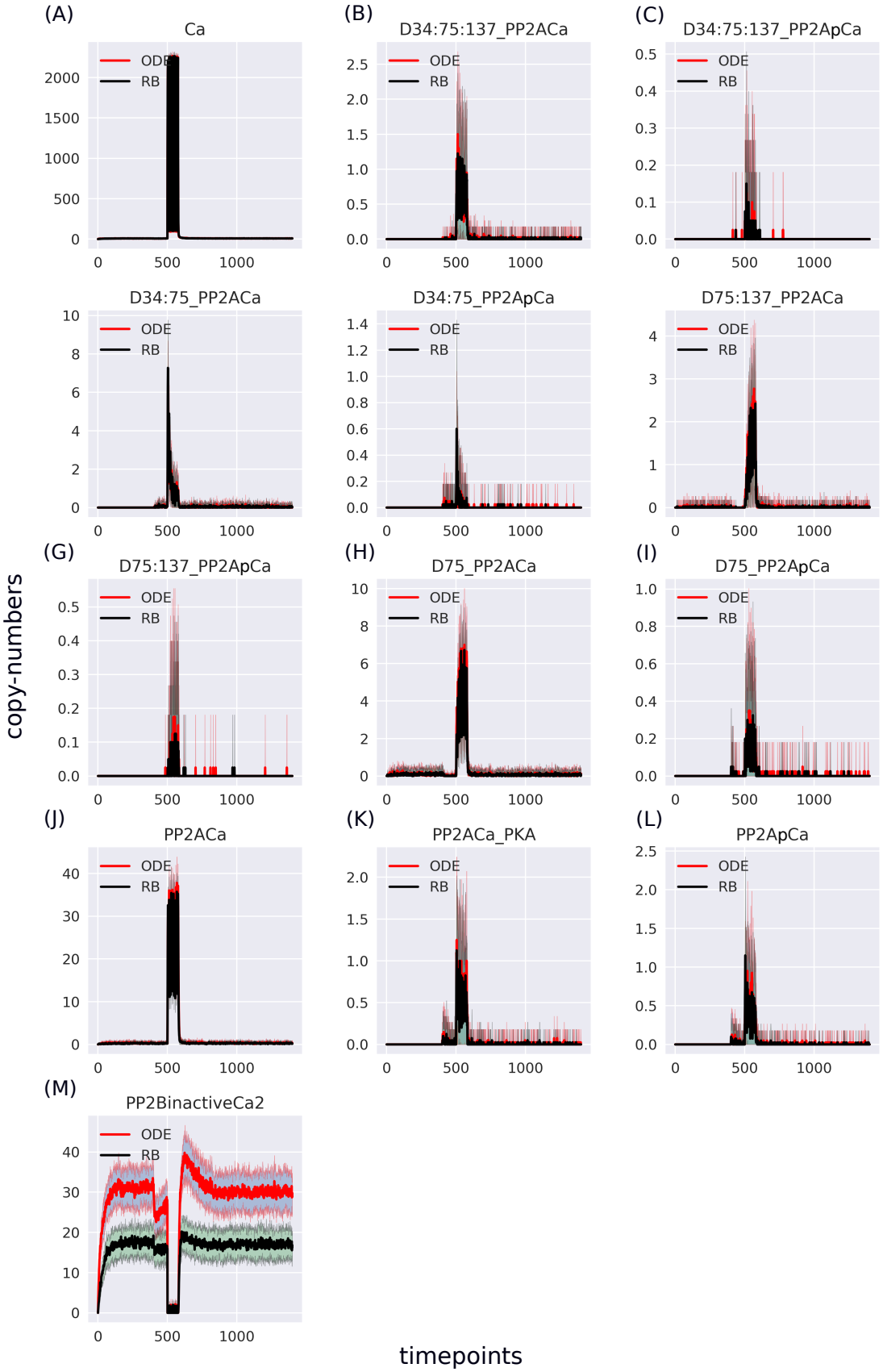


FIGURE 2.17: Separated molecular species containing Ca^{2+}, selected as in the original model. The trajectory of "PP2BinactiveCa2" of the RB model was obtained by selecting one of six entities representing an inactive form of PP2B in the RB model. A discrepancy between models is still present, but this time the trajectory is lower for the one obtained with the RB model.

"PKA*" observable also reaches a much lower peak than its ODE counterpart. Accordingly, values of rate constants of rules that number increased due to the "combinatorial binding" notation in the RB model should be increased to closely match the ones in the ODE model.

The above analysis is an example of how the RB modelling is a flexible tool to explore complexes of species that appear during the simulation. Molecular species defined in the ODE modelling belong to a fixed and definite component of the model, whereas in the RB modelling, molecular species that are formed during the simulation is a very much a subject of investigation. By using samples of molecular mixtures obtained with snapshots, it can be determined what molecular species are created and in what abundances. RB model simulation results can be easily dissected the very fine detail by choosing observables of interest to be tracked during the simulation. In the perspective of reusing a model, the automation of collecting agglomerated time courses of observables of interest renders the RB modelling as more advantageous in comparison to ODE-based modelling. This automated procedure of defining observables is less prone to errors and independent of variable names as it happens in the ODE model.

On the other hand, this detailed and explicit notation offered by Kappa might appear redundant as exemplified by enumerated locations of four Ca^{2+} ions on four indistinguishable sites of PP2B. This necessity to indicate all possible combinations of sites is a particular drawback of the Kappa language. The other one is time necessary to simulate the model. In general, the length of simulation of stochastic models, compared to deterministic ones, has been always an issue addressed by multiple optimisation strategies [161]. It is no different in case of the two compared models. It takes almost 40min[4] to simulate this particular RB model with the KaSim simulator. Solution of the ODE model in the COPASI environment in the deterministic setting is returned in an instance. As the COPASI stochastic simulator is able to produce results in no more than 15sec, this difference in simulation time does not stem from the nature of the stochastic simulation. This fact largely limits the use of KaSim and ought to be considered in decision which formalism is most advantageous for a particular problem.

[4]The total CPU time measured with the "`time`" command on Linux OS.

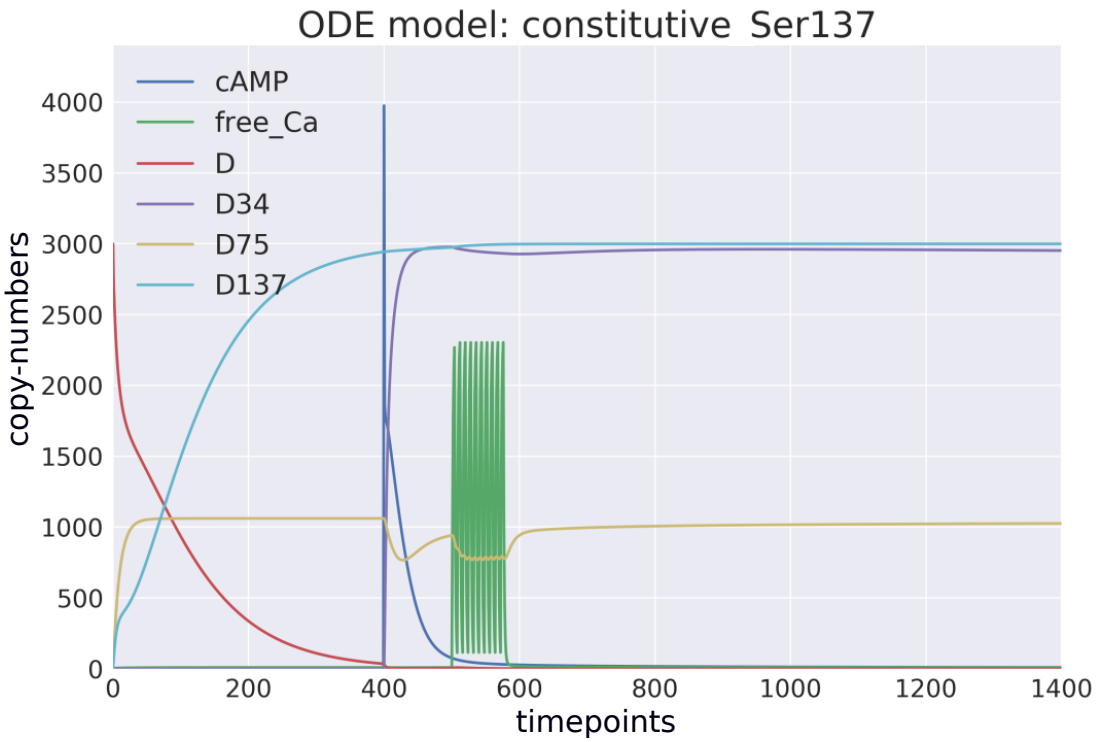
2.4.2.1 Comparison of models with site-directed mutations

Up until this point, we saw the outcome of comparison performed on models in the wild-type condition. The authors of the ODE model analysed it in two other conditions of site-directed mutagenesis affecting the Ser137 site. The same type of perturbations, based on modifications of constant rates, were applied to the RB model to establish if this perturbation would generate similar results, and to obtain a broader view on differences between models. Figure 2.18 juxtaposes simulation results of both models affected with the constitutive Ser137 mutation. As exemplified by the six key observables showed in this figure, there is a close match in initial conditions and general pattern of dynamics between time courses of the two models. During the first minutes of the simulation in both models, the Ser137 site completely depletes the unphosphorylated DARPP-32 as the site cannot be dephosphorylated. DARPP-32 phosphorylated at Ser137 reaches the maximal abundance level of 3000 copy numbers. At the 400th second, the cAMP pulse is injected causing phosphorylation of DARPP-32 at Thr34 that reaches the same maximal level as DARPP-32 phoshorylated at Ser137. In the wild-type model variant, phosphorylation of Ser137 significantly slows down dephosphorylation of Thr34. This occurs because the catalytic rate constant in the dephosphorylation reaction of Thr34 by PP2B is 133 fold lower when Ser137 is also phosphorylated. Therefore, this persistent phosphorylation of Ser137 strongly inhibited dephosphorylation of Thr34 locking the effect of the cAMP pulse such that Ca^{2+} spiking had marginal influence on the "D34" trajectory. A close match between models is confirmed by a view on similarity between four paired curves in Figure 2.19. Selected observables are the only ones among 15 that demonstrate the impact of the mutation. Three of these four observables have been earlier mentioned as bearing the largest discrepancies between models (Figure 2.19A, C, D), where the copy-numbers at steady states in the RB observables are slightly lower.

The other site-directed mutagenesis is the Serine to Alanine mutation. The results of comparison can be viewed on Figure 2.20. This mutation has more subtle effect. Its major result is a deeper plunge of "D34" induced by the Ca^{2+} stimulus and a slightly lower second peak in the relaxation phase than in the "wild type" model variant. The same effects can be viewed in RB model results.

Based on these two examples of site-directed mutations, the same per-

(A)



(B)

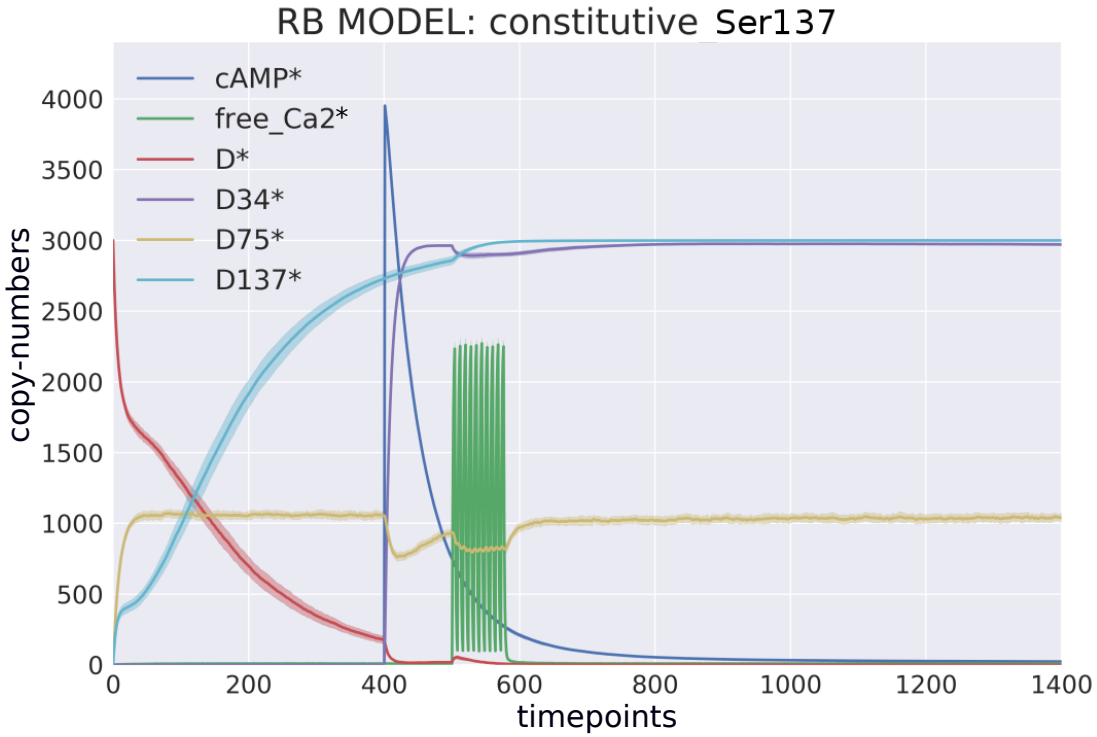


FIGURE 2.18: Comparison of constitutive Ser137 mutation induced in (A) ODE model in deterministic setting; (B) RB model in stochastic setting. The same ingeration performed on rate constants of the two models caused similar dynamics.

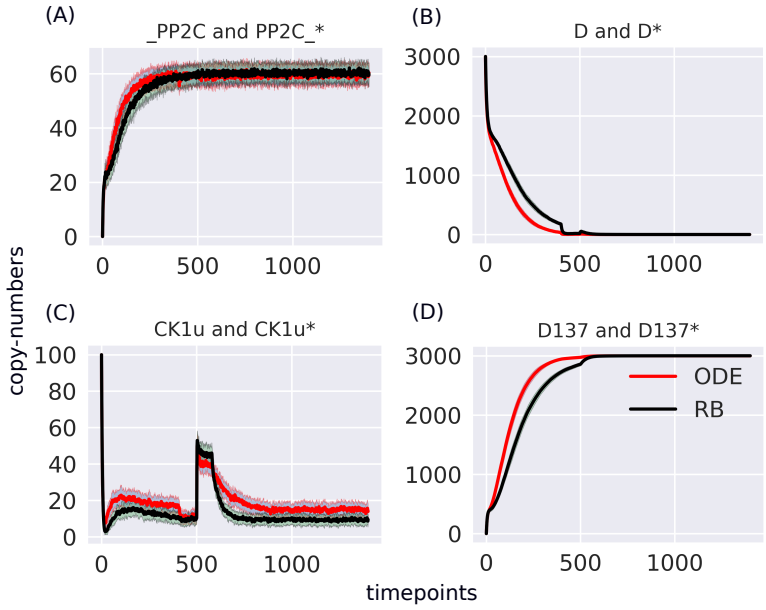


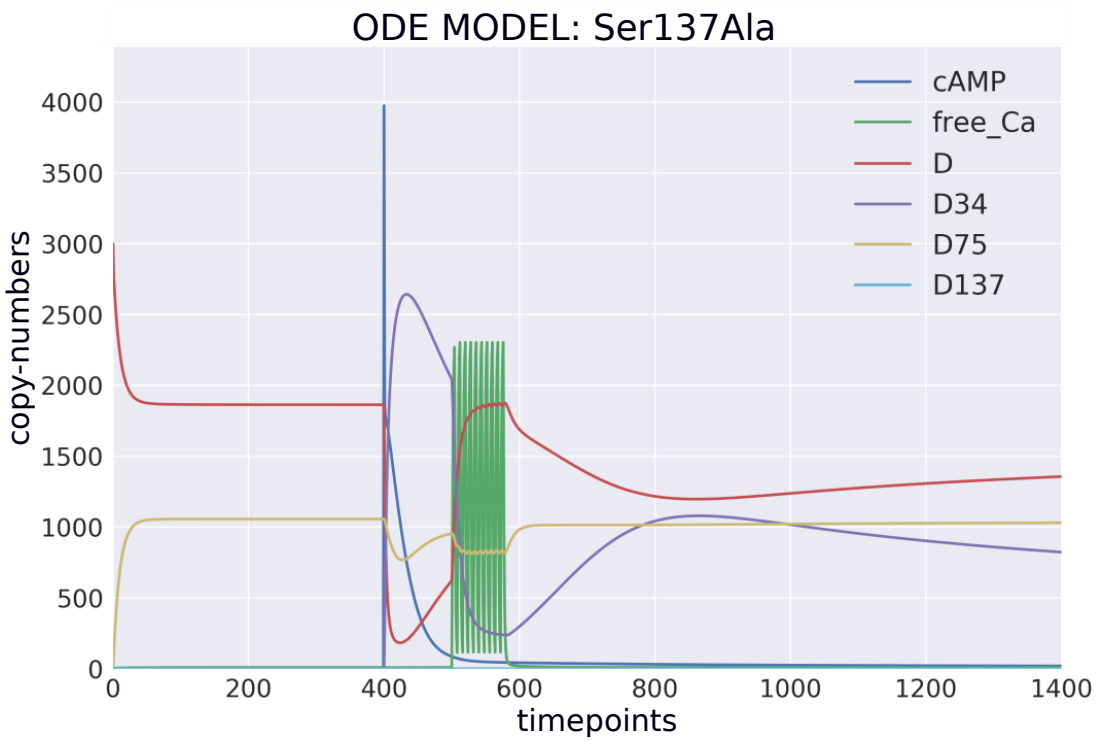
FIGURE 2.19: Closer look at traces of observables affected by the constSer137 mutation. Both models were simulated in the stochastic scheme.

turbations used in the ODE model were successfully applied on the RB model producing similar dynamics. It means that the RB modelling allows to emulate experimentally observed perturbations in a similar manner as it is performed in the ODE-based modelling.

2.4.2.2 Comparison of trajectories between competitive and non-competitive variants of rule-based model

As discussed in *Section 2.3.2.1*, modifications of molecular binding interfaces resulting with new complexes would require to enumerate additional molecular species and extend the model with additional equations in the ODE-based framework. Contrary to this, the RB language allows for easy modification of agents' binding properties. Because DARPP-32 is an intrinsically disordered protein and the way it binds to partners is yet to be known, two RB models with different variants of binding site specifications are compared to test if dynamics of the RB model would be affected if DARPP-32 could bind more partners at the same time. The first one, as defined in the original model, constitutes a competitive version of the model with one binding site of DARPP-32 (oBS). The second constitutes a noncompetitive version of the model where partners of respective phosphorylation sites can bind concurrently (three-binding-sites DARPP-32, tBS). This modification was made

(A)



(B)

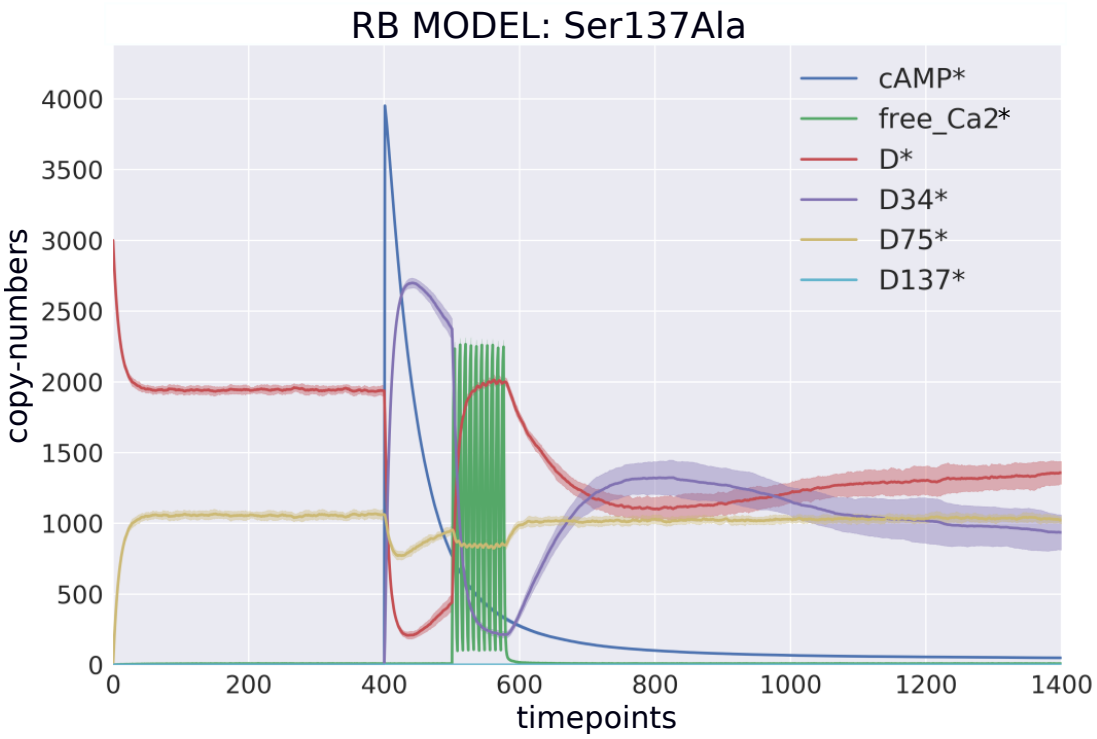


FIGURE 2.20: Comparison of the Ser137Ala mutation (A) ODE model in deterministic setting; (B) RB model in stochastic setting.

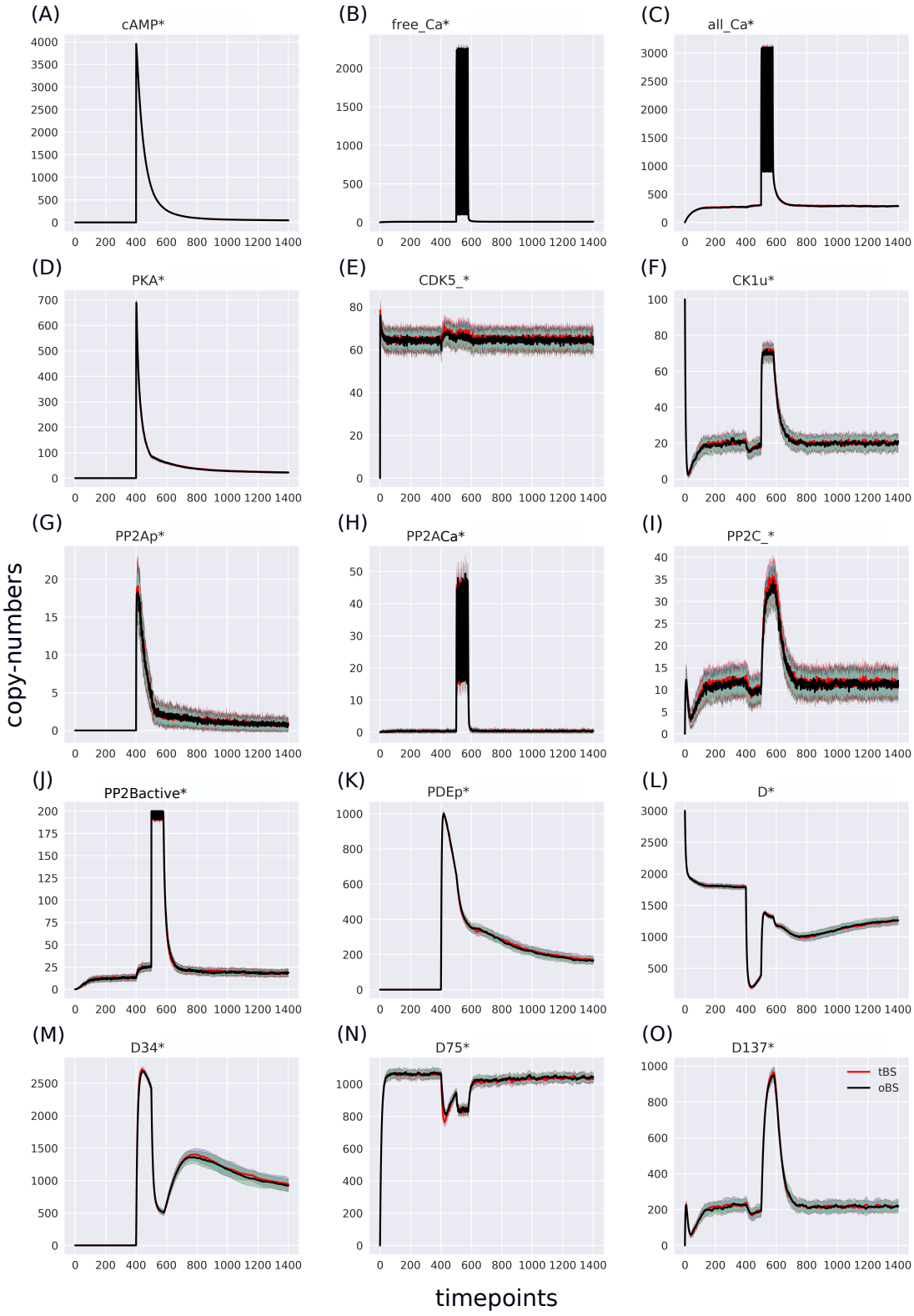


FIGURE 2.21: Comparison of two RB model variants where the agent representing DARPP-32 had one binding site (oBS, red trace) and three binding sites (tBS, black trace). Overlaid trajectories of corresponding agents demonstrate no effect of this modification on the model trajectories.

by changing the interface of DARPP-32 in the agent specification, and rules defining behaviour of DARPP-32. The trajectories of two models were superimposed in pairs of corresponding observables. Figure 2.21 demonstrates that the modification of binding site properties has no effect on trajectories of observables. A direct consequence of this modification was an increase of complex sizes to more than two proteins. Therefore, it seems that larger complexes are potentially rare to be formed during the simulation. An example of such complexes can be seen in Code 2.13.

Code 2.13: Examples of DARPP-32 complexes composed of more than two proteins

```
# One phosphatase and one kinase:
D(ser137~u,thr34~p!0,thr75~u!1),CDK5(a!1),
PP2B(ca1!2,ca2!3,ca3!4,ca4!5,state~a!0),
Ca2+(a!4),Ca2+(a!3),Ca2+(a!2),Ca2+(a!5)

# Two phosphatases :
D(ser137~p!0,thr34~p,thr75~p!1),PP2A(ca!2,pSite~u!1),
PP2C(a!0),Ca2+(a!2)
```

However, as mentioned in *Section 2.4.1*, the difference between one-binding-site DARPP-32 (oBS) and models in the total number of molecular species created during the simulation is quite considerable. There are 46 more molecular species formed in the tBS (137) than in oBS (91). To examine the increase of newly formed complexes of the simulation in the non-competitive model variant compared to the competitive one, counts of unique molecular species per timepoint are obtained from snapshots (described in *Section 2.4.1*). In Figure 2.22A, we observe that the species set size is quite similar in both model versions. Furthermore, the dynamics of forming complexes is dictated by the stimuli input pattern (Figure 2.22B). The largest differences between oBS and tBS also take place during the stimuli application.

The increase in the number of binding sites of DARPP-32 resulted with unchanged dynamics. It should be noted that this was only established for a single setting of parameters and initial conditions. As these three binding sites do not counter each other's binding properties, this lack of difference might be caused by similarity in occupancy between a single site and all three sites together. The probability of a site to be bound depend on copy numbers of

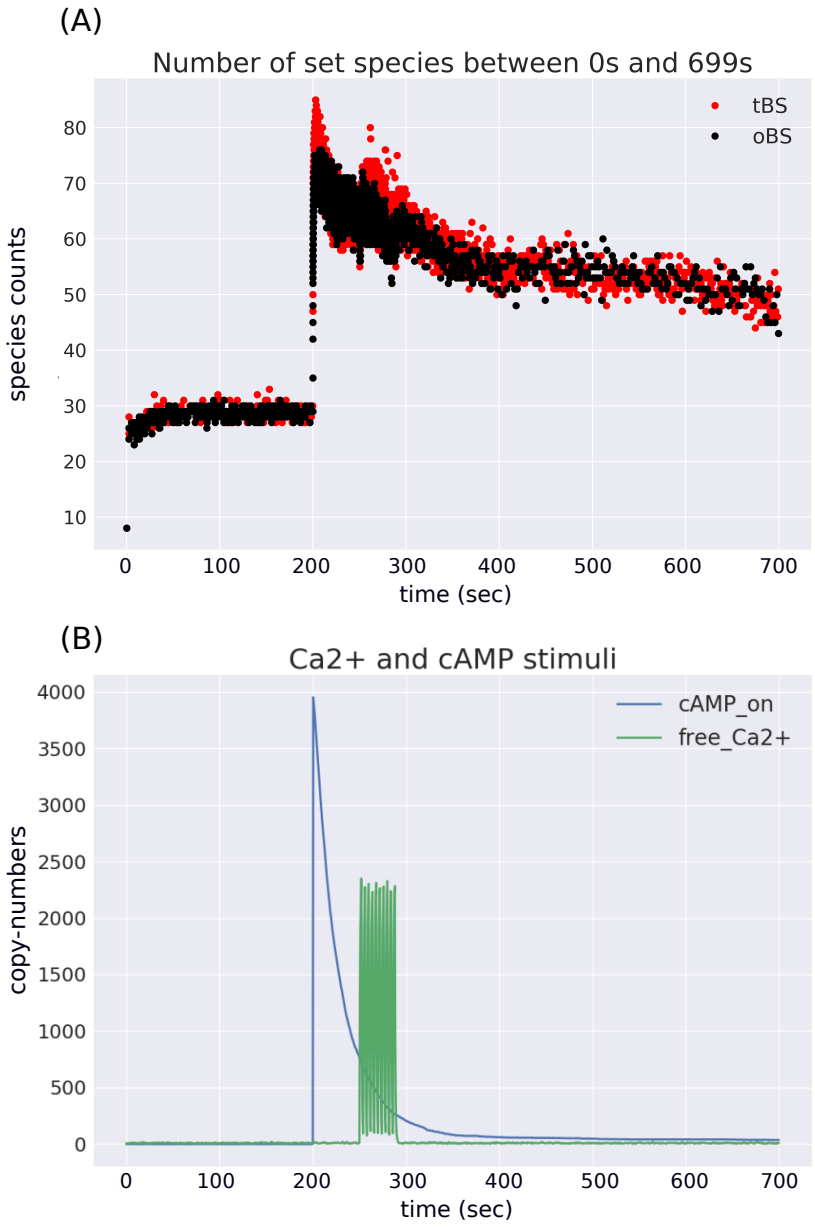


FIGURE 2.22: Overlay of change in species counts over time for one-binding-site DARPP-32 (oBS) and three-binding-sites DARPP-32 (tBS) model variants (A). The change in the number of unique species counts is aligned with trajectories of the stimuli (B).

reactants and strength of binding affinities. Reactions in the model are classified as weak, with dissociation rates in the range of μM. Low affinity bindings generally lead to lower levels of site occupancy. Moreover, the amount of DARPP-32 molecules exceeds the total counts of all its interactors. Therefore, with the current proportions of reactants, all three sites of DARPP-32 cannot be saturated to expose the difference in binding capacity of DARPP-32. To expose potential differences in dynamics between two binding scenarios, the site occupancy should be targeted. The simplest test of this explanation could be achieved by altering the size of reactant pools. For instance, a significant decrease of the levels of DARPP-32 could increase the proportion of other reactants. A prediction could be made that a similar dynamic response of DARPP-32 could be achieved with its lower copy numbers.

2.5 Discussion

ODE-based modelling is a classical and most commonly used method for creating detailed dynamic models of biological systems on many levels of biological organisation [222–224]. It is often a point of reference or comparison to newly proposed modelling methods [108, 136, 147, 225, 226]. Nevertheless, modelling of signalling systems with ODEs poses particular difficulties due to complexities underlying biomolecular interactions in cellular signalling that occur transiently between multiple partners, forming complexes composed of molecules that can exists in more than one functionally different state [30, 150]. The RB modelling was developed to address these difficulties by using formalisms originating in computer science, in particular graph transformations and process algebras. Although a large body of reviews discussed advantages of the RB modelling over the ODE modelling [106, 120, 124], to the author's knowledge, none of these studies presented a direct comparison of time courses of one molecular system encoded in the two modelling formalisms. This chapter presents a process of encoding reactions underlying an ODE model to the RB language and comparison of the two models with respect to model specification and simulation results. This can provide a clearer view of differences between the two formalisms and allow to answer such question as:

- how does encoding reactions into rules affects a system dynamics compared to encoding them in ODEs?

- does specification of a model in the RB language can facilitate process of understanding and reusing dynamical models?

- what kind of biological processes are most advantageous to model in the RB framework?

In the first step, models specifications were divided into components such as reactions or rules, molecular species, and rate constants. Then, corresponding components were compared with respect to their overall counts. Encoding reactions into rules slightly reduced the size of model specification and increased counts of molecular species. This result confirmed a well known advantage of encoding chemical reaction networks into rules. Translation of reactions into rules involves identification and removal of non-influential context carried by reactants, such as their phosphorylation or binding state. Therefore, by preserving only the context that conditions reactions allows to rewrite these reactions into a generalised pattern, what in consequence shortens and simplifies model representation.

As the drop in rule number was rather low, reactions were closer analysed by dividing them into subsets representing more general molecular mechanisms, such as activation and phosphorylation. This showed that reduction in the reaction number is only true for reactions occurring between the same reactants, describe the same transformation, and are parametrised with the same values of rate constants, but differ only with respect to binding or internal state of reactants. In this type of reactions, the number of unique reaction rates was equal to the number of rules that reactions were rewritten to. This is an example of tackling combinatorial explosion by the RB modelling where multiple reactions, parametrised with rate constants of the same value, are equivalent to a single rule pattern.

Increase in the number of rules representing reactions occurred in cases where the same partner binds an agent at multiple sites. To encode such reactions, a much larger number of rules had to be encoded to represent all possible positions and stages of binding process. This "combinatorial binding" notation is not a general property of the RB language but only characteristic to the Kappa syntax. In the BioNetGen Language (BNGL), an alternative RB-language to Kappa, a rule can be defined with sites named in the same way. It implies that a rule pattern defined for one of them applies to the others

[154]. However, it would have to be empirically established how this simplified representation in BNGL would affect trajectories of observables.

In the next stage of models comparison, trajectories of corresponding observables were analysed. This shown an overall agreement between models dynamics. By overlying 15 observable pairs, one from each model, discrepancies between time courses were observed. The "all_Ca" observable was examined closer as an example of such discrepancies. As molecular species are not defined in the RB model specification, the exact composition of created molecular species was obtained by taking snapshots of molecular mixture during the RB model simulation. We learned that there are more molecular species with Ca^{2+} in the RB model, than in the ODE model (24 vs. 13). Closer examination revealed that 6 molecular species of the RB observable were not included in the ODE observable as their name did not contain Ca^{2+}. Of the total of 24 species, 18 represent exactly 13 species of the ODE model. This disproportion in the number of molecular species was due to expanded representation of half-active PP2B, where two Ca^{2+} ions can occupy 4 sites of PP2B in 6 variations. When the simulation of the RB model was performed with observables consisting of exactly the same species that composed the "all_Ca" observable in the ODE model, it appeared that all paired trajectories matched perfectly except for the half-active PP2B. In the RB model, this observable is a sum of 6 variants of molecular species that had much higher abundance than its ODE equivalent. Interestingly, abundances of a fully active PP2B were lower for the RB observable than for the ODE. A fully active PP2B is one of the two observables that activation in rules required the "combinatorial binding" notation that specify all combinations of binding positions of Ca^{2+}. The second one was PKA. Pairing ODE and RB trajectories of PKA demonstrated that this observable is also present at the lower abundances in the RB model simulation. This observation suggests that in the RB model simulation, activation of proteins encoded with the "combinatorial binding" notation are much slower processes than in the ODE model. What follows then, these processes should be parametrised differently if one wanted to obtain a perfect match to these two trajectories of the ODE model. The observed discrepancies, specific to this particular group of rules, ought to be taken into account when reactions and rate constants of ODE models are reused to construct RB models.

A potential cause of such discrepancy between reactions and rules in

terms of reaction speed could not be attributed to differences between the RB model and the original model simulation procedures as both were simulated with variants of the Gillespie's Stochastic Simulation Algorithm (SSA). During one iteration in the RB model simulation, a rule pattern is applied to one rule instance found in the molecular mixture over time, what is called as an "event". Therefore, agents that activation is expressed with larger number of rules, such as these with the "combinatorial binding" notation, require more events that trigger these rules to be activated. This might have caused the lower abundances of PKA and the active PP2B in the RB model, compared to the ODE one.

Presented dissection of the "all_Ca*" observable with application of snapshots demonstrated how one can explore in detail emerging molecular species during the simulation of an RB model. Overlying trajectories of molecular species that compose "all_Ca*" and "all_Ca" observables allowed to track the source of differences between models. This detailed description of reactions on the subprotein level offers precise view on interaction details. Emerging molecular species are tractable, relatively easy to examine and analyse. Contrary to these observations, the ODE model heavily relies on the complete knowledge of the system and the model encoding.

Observables of the ODE model were obtained by identification of molecule names in variables representing molecular species. This simplistic approach omitted Ca^{2+} ions bound to PP2B and multiple copies of cAMP in complexes. Therefore, retrieval of molecule counts hidden in individual molecular species of the ODE model would require deeper deconstruction of the reaction system to reuse and explore the model in an automated manner. Therefore, difficulty in identification of molecular entities among molecular species impede also correctness of reusing ODE models.

A simpler approach to correct identification of molecules in species could be potentially performed by parsing the SBML file, that the ODE model is also encoded with. This could be done because the model is deposited in BioModels, a database of mathematical models, and it was successfully curated. The curation process guarantees annotation of molecular species with common resource identifiers, as defined by the Minimal Information Requested In the Annotation of biochemical Models (MIRIAM) standard. In the Fernandez

et al. [177] model web page in the BioModels website[5], under the "Physical entities" tab, it can be observed that an active PP2B is a composite species having Ca^{2+} ions and therefore, this molecular species is annotated with two common identifiers, one for the protein, one for the ion. However, based on this information one will not learn that there are in fact 4 Ca^{2+} ions. Lastly, the two identifiers are not always present in other complexes of PP2B (e.g. CK1) although an active form of PP2B in this model denotes a protein complex with ions. It can be concluded that automated identification of molecules in the RB framework by defining observables is particularly advantageous, as it offers a transparent framework with error-prone identification of molecules in a modelled system. This feature is particularly vital when large number of molecules is involved in the modelled system and their particular states are important to be analysed in detail.

The model was tested with two types of site-directed modifications affecting one of the phosphorylation sites of DARPP-32. The first one, common for ODE, was based on a parameter modification and reproduced from the original study. The results of the RB model matched closely the ODE model in all conditions proving that the same method commonly used to perturb the ODE models can be applied to the RB model. This is a particularly advantageous regarding flexibility of a modelling framework to reproduce experimentally conducted perturbations.

The second modification type was based on alteration of the DARPP-32 binding capabilities. This modification, on the other hand, is only specific to the RB domain. Two variants of the model were introduced with varied numbers of binding sites of DARPP-32. This binding site modification, that effectively changed the model reaction network, did not affect the model response. This might be caused by similarity in the average site occupancy between the model variants in the current model conditions that could be altered by changing proportions of reactants and binding affinities..

There are at two main routes alongside which the RB model could be further explored. The first one is modification of parameters defining different phases of combinatorially bound Ca^{2+} ions to PP2B, and cAMP to R2C2. A particular task would be to establish factors by which binding constants should be changed to counterbalance the larger number of intermediate variants of

[5] http://www.ebi.ac.uk/biomodels-main/BIOMD0000000153

these complexes and lower copies of their final activated forms. The other aspect worth exploration in future is to identify conditions under which a difference in dynamics caused by increase of the number of parallel binding sites of DARPP-32 can be observed. As a starting point, the simplest modification could be achieved by significantly decreasing the copy numbers of DARPP-32 compared to other interactors. As mentioned by the authors of the Fernandez et al. [177] model, levels of DARPP-32 vary significantly between μM to tens of μMs in the striatum. With the greater availability of single-cell techniques for quantification of protein numbers [227] it would be worth to establish more precisely the range of DARPP-32 even on the resolution of a dendritic spine [228]. Estimation of variability between cells could also be used to compare the level of noticeable variability of DARPP-32 phosphorylated at Thr34 observed between repeated simulation runs in the stochastic framework.

Defining a model with rules allows to encode details of molecule internal states, binding sites and track changes in abundances of resulting molecular species. Therefore, the process of encoding rules turns attention to questions such as how many binding partners can bind a protein at the same time. Interestingly, this question was not discussed in the Fernandez et al. [177] study regarding the DARPP-32 interaction network. The translation process has shown that information about interfaces of interacting proteins and their alternative states seems not only natural to include in a model build in the RB framework but also would largely ease the process of its development as it could support decision on agents signatures. One of the most studied protein states happen through changes in post-translational modification (PTM) sites, among which protein phosphorylation is most common [229, 230]. Information about protein interaction interfaces can be based on protein domains which are important functional protein units mediating interactions. For instance, proteins containing phosphatase catalytic domains are enzymes of dephosphorylation reactions [229]. Therefore, among low-level data resources that could support the RB modelling an important position can be allocated to resources of protein domains and PTMs.

2.6 Conclusions

In this chapter, I showed that the RB model recapitulates the general dynamics of ODE-based model. Analysis of the RB model in comparison to

the ODE, proved its expressive and flexible syntax for encoding and studying crucial aspect of signalling networks, such as complex formation. RB modelling provides a framework for testing impact of perturbations on the reaction network as well as offers tools for exploration and dissection of emerging molecular species during the simulation. However, increasing number of such unpredicted molecular species of unknown importance might become intractable. Particularly, in such commonly used methods of parameter analysis to identify ones that have the largest impact on the model output. Selecting such model output might not be a straightforward task when molecular species are created in the simulated system. It would be advantageous to support modeller's assumptions and knowledge about the modelled system with automated methods to differentiate between such species. Therefore, in the next chapter a proposition of a framework for prioritisation of the RB model output is presented.

Chapter 3

Observable prioritisation and global sensitivity analysis for rule-based models

3.1 Motivations

It is a common practice when modelling molecular systems to reduce the model simulation outputs to a handful of readouts that summarise some specific characteristic of the response of interest [231, 232]. This simplification is a crucial step to progress with the model analysis, such as visual examination of the simulation output. Selection of biological readouts is commonly based on the modellers knowledge of the system and the availability of experimental data against which to compare model performance. This might become difficult with the increase of model sizes, gradual automation of model construction and when some aspects of modelled behaviour are impossible to experimentally observe. What is more, new methods of modelling, such as RB modelling, that show a different perspective on modelled systems, might require formal approaches to support the knowledge of a modeller to select appropriate model readouts.

As demonstrated in *Chapter 2*, a rule-based (RB) model is defined with a rule patterns that are applied to multiple reaction instances. Therefore, in contrast to the ordinary differential equation (ODE) modelling techniques, a list of molecular species produced by such reaction instances is not defined *a priori* in RB models but created over the simulation. Each simulation is also a

sample of all possible molecular species that can be created and interact in the system. Therefore, as opposed to ODE, the composition of species populating the simulation requires new means to be carefully studied. If RB models generate a relatively large number of molecular species, it might become difficult to decide which of these species are important and therefore, used as model readouts. It would be particularly important to analyse changes in composition of molecular species during the simulation with respect to variations in model basal conditions designed to emulate disease conditions or drug-induced perturbations. For instance, by modelling and identification of shifts in importance of interacting molecules, their configuration of states or involvement in complexes, might help to study molecular mechanisms underpinning resistance to pharmacological interventions observed in targeted cancer therapies [233]. In particular, how drug targeted pathway signal can be bypassed by rewiring intracellular signalling in tumor cells as an adaptive stress response [234].

Selecting only a subset of created species is a crucial step in the model analysis for example in application of sensitivity analysis (SA). SA is an approach to quantitatively evaluate how a change in model input parameters can affect the model simulation output. It is a crucial step of every modelling task as there are large number of sources contributing to uncertainty in parameter values in mathematical models, such as difficulty in their accurate experimental determination and different experimental conditions they were measured in [231, 235]. SA can help to answer such questions as: which parameters reduce uncertainty in model output, which parameters do not contribute to its variability, and which parameters have the largest effect on output under a particular model setting [236]. Different simulation outputs can be sensitive to change in different parameter subsets. Therefore, if we were able to cluster the simulation outputs into subgroups that are strongly interlinked, then we could separately subject each group to SA, that would identify parameters that are most important to each subgroup of simulation outputs.

In this chapter, a pipeline for extended and automated analysis of RB simulation results is proposed (Figure 3.1). It is performed on time courses obtained from simulation of the RB model presented in *Chapter 2*. Main components of pipeline are: a method for clustering and prioritisation of time courses, followed by application of SA to calculate sensitivity scores of parameters with respect to the chosen subset of simulation outputs. The particular

choice of SA method was guided by its potential application to larger and more complex RB models than the model of DARPP-32 network and therefore, is scalable and model-independent. Prioritised model outputs and parameters are jointly presented as a network with edge weights derived from sensitivity scores. The choice of network representation has two particular justifications. First, as SA is performed on more than one model outputs, a network structure can facilitate examination of relations between groups of selected observables and parameters. Second, common experimental design in modelling studies involves application of perturbations to a baseline model and examination of outcomes of these perturbations in reference to the baseline state. Investigation of differences between different modelled conditions can be facilitated by representing them as networks as key simulation outputs and control parameters are unified and summarised in a compact data structure. In particular, measures contained in two networks representing different model conditions are subtracted from each other. By taking the difference between two model conditions, one can expose changes in sets of control parameters and observables induced by perturbation in the model. In such setup, different graph-based techniques can be applied to analyse such networks. To investigate this approach, the pipeline is applied to two model phenotypes presented in *Chapter 2*: the model with a baseline setting (*wild-type*), and the model with constitutive Ser137 (constSer137) site mutation (*perturbed*).

This approach aims to extend the knowledge derived from analysis of complex output of computer generated models that could facilitate and guide laboratory experiments thus tightening interactive feedback loop between computational modelling and experimental work.

3.2 Introduction

The pipeline for automated analysis of RB models is divided into three stages (Figure 3.1). Each stage involves methods and tools that were previously developed and applied in different studies. The first stage involves selection of model simulation outputs. In the context of RB modelling these outputs are trajectories of observables that are recorded during the simulation. The second step identifies parameters that variation has the strongest impact on selected observables. In the last step, results obtained in two previous steps are integrated into a network representation.

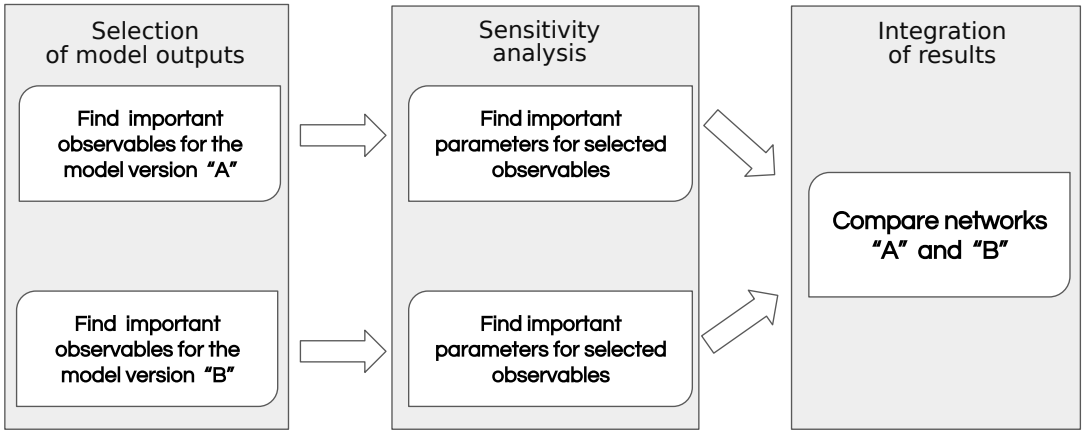


FIGURE 3.1: Outline of the pipeline steps to automate the analysis of RB model simulation results.

Before directly delving into details of the pipeline, each method composing the pipeline is presented with the justification of choice. The method for selection of model observables is introduced in *Section 3.2.1*. *Section 3.2.3* flesh out details of selecting a method of SA that is most appropriate for RB modelling. Next, in the methodology section (*Section 3.3*), the pipeline will be presented as a whole procedure (*Section 3.3.1*), followed by explanation of approaches and decisions required to be taken to perform methods used in each of the pipeline step. The methodology section is followed by presentation of results from comparison of networks derived from the "wild-type" and "perturbed" model simulations (*Section 3.4*).

3.2.1 Clustering and prioritisation of observables

Identification of communities or clustering was earlier discussed in the context of network analysis (*Section 1.3.1*). In the field of data mining, clustering is a common data pre-processing step [237]. It is an exploratory technique that partitions the data set into clusters, that members are highly similar with respect to predefined objective. Dividing variables into similar subgroups reduces the complexity of the data set as smaller variable subgroups are easier to analyse in separation. Clustering is one of unsupervised classification methods that partitions unlabelled data into clusters by learning from observations rather than learning from examples of correct answers [238]. This type of classification is particularly important aspect for this study as there is no correct examples that define a structure in the data composed of observable trajectories

obtained with the simulation of a RB model.

There is no single formal definition what is a cluster and therefore, there is a great variability between methods. The most popular methods of clustering are variations of *k*-means, hierarchical, density-based, model-based and graph-based clustering. The successful application of any method depends on the data. Therefore, the choice of an appropriate clustering method should be guided by data characteristics. Andreopoulos et al. [238] defines evaluation criteria for clustering methods in a biomedical context: scalability, robustness to outliers, insensitivity to the input ordering, minimum user-specified input, mixed data types, arbitrary shape of clusters, insensitivity of results to duplicates in data. In general, molecular dynamic models of signalling systems tend to be complex, composed of different feedback and feedforward loops, manifesting non-linear dynamics. Efficiency and scalability of clustering method are criteria that can eliminate a method from the very start. The scalability criteria is important aspect in this study if we consider to include all molecular species produced during the simulation. As seen in *Chapter 2*, the number of molecular species in the RB model of dopamine- and cAMP-regulated neuronal phosphoprotein with molecular weight 32 kDa (DARPP-32) network generated 91 different molecular species in the competitive variant of the model in which DARPP-32 can only bind one partner at a time. In its non-competitive variant, this number was 137. With the increase of complexes and combinatorial mixtures of species it will be crucial to constrain the choice of clustering method to the most efficient and scalable one.

Clustering methods using information theory measures appear to be most appropriate for high-dimensional and complex patterns in data [239]. Information theory is a study concerned with quantification of information content. Clustering methods based on information theory use measures extended from mutual information introduced by Shannon [240]. Mutual information quantifies dependency between random variables. In information-theoretic clustering is based on maximisation of the mutual information between data points and cluster labels [239]. The information theoretic approach offers a formal and operational method for clustering that does not introduce a prototypical cluster nor make assumption about the data distribution [239] and it addresses such critical issues as sensitivity to outliers, present in commonly used methods [241].

Information theoretic frameworks for the clustering problem are in constant development for finding a common theoretic ground and that success has been mainly based on the evaluation against experimental results [241]. Still, it is a quite dynamic field [242, 243] with examples of applications in the biosciences [241, 244].

For this study, Correlation Explanation (CorEx) was chosen as a highly optimal method that leverages information-theoretic objective. From the point of computational complexity, the algorithm scales linearly with respect to the number of variables. Moreover, the algorithm can be parallelised on multiple cores that speeds up the procedure.

Regarding the minimum user-specified input, the CorEx algorithm requires input parameters to be specified by a user, such as the number of hidden variables representing clusters. If this value is too high then only a subset of clusters will have associated members [245]. Therefore, this number can be optimally determined by starting from a relatively large value to the size of clustered objects, e.g. smaller than their number, and then gradually reduce the number of clusters up to some margin. Furthermore, the information-theoretic objective used in CorEx is insensitive to missing values or noise [245]. One major potential drawback of CorEx is that it does not arrive at the global optimum. Therefore, the analysis of results should be based on multiple runs of CorEx.

CorEx has been mentioned by other studies but it has been systematically compared with other methods by the authors. As oppose to other alternative methods, the authors showed that CorEx was able to partition a synthetic data set with perfect accuracy in spite of a gradual increase in the number of variables (Figure 3.2A). The method was also tested on a real data set, the Big-5 personality test, and as the only one resulted with a correct answer (Figure 3.2B).

The comparison of clustering methods is a nontrivial task as it requires a selection of data sets, definition of correct clustering result and parameter choice for each tested method. Wiwie et al. [246] compared 13 most popular methods with 24 common bioinformatic data sets but did not include methods that are based on information theoretical objective. The largest data set they used was from gene expression measurements in bone marrow comprising 999 genes and 38 samples [247]. Gene expression across tissue samples

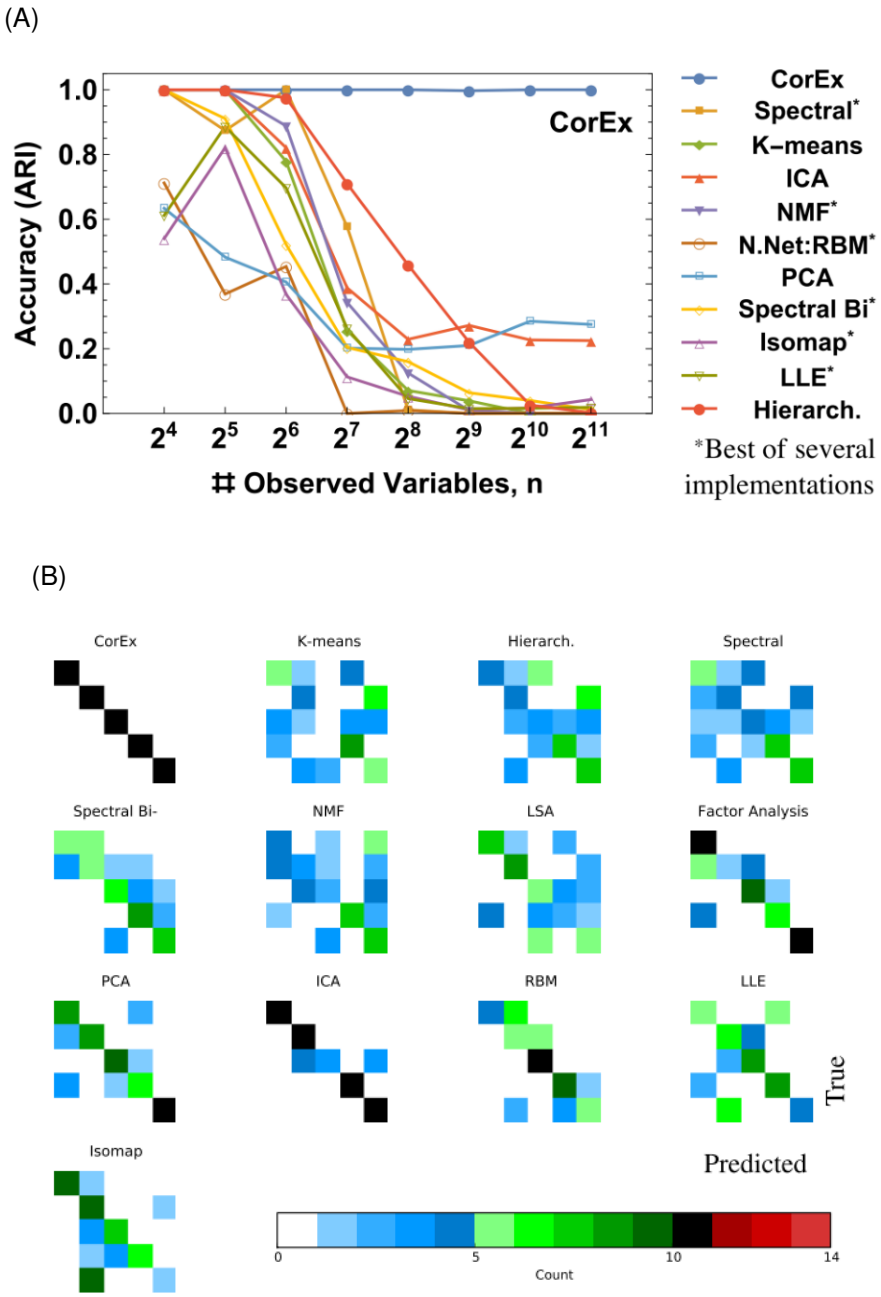


FIGURE 3.2: Comparison of CorEx to other clustering methods performed by by the authors. (A) Comparison of partitioning accuracy based on Adjusted Rand Index (ARI) (see *Section 3.2.2*) with respect to increased number of variables. The data were synthetically generated with a latent tree model. As oppose to other methods, CorEx retrieved all clusters despite a gradual increase in the number of observed variables. (B) Clustering results of the Big-5 personality test data set obtained with 13 methods (CorEx+12). Results are presented as confusion matrices that compare predicted to true clusters. CorEx correctly divided the data into 5 clusters. Source of figures: Ver Steeg and Galstyan [245].

is particularly challenging due to undersampling. The number of samples is much smaller than the number of variables. CorEx has also been applied to gene expression data from ovarian tumors but this data set was much larger comprising 5371 genes across 420 tissue samples [248]. Pepke and Ver Steeg [248] compared the results with outcomes from two most popular clustering methods, *k*-means, hierarchical, and a method for dimensionality reduction, Principal Component Analysis (PCA), showing the advantage of CorEx. In another biological context, CorEx was used to cluster mixed data types composed of different biological measures to search for the most informative biomarker of cognitive decline and brain atrophy [249].

A decision to include CorEx into the pipeline presented in this chapter is supported by results of a preliminary study where CorEx was applied to a similar purpose and type of datasets, composed of multivariate time courses obtained from simulations of RB models published in Suderman and Deeds [123] (*Appendix C*).

3.2.1.1 Correlation Explanation method

The authors of CorEx link the algorithm with multiple domains of machine learning, placing the method in a context of hierarchical representation learning and dimensionality reduction of high-dimensional datasets [250]. In more precise words, CorEx is an unsupervised learning method for finding a hierarchical structure of *latent variables* based on dependencies in data. Latent or *hidden variables*, are variables that are not directly observed but inferred from the relations between variables that are measured. In this context, measured variables are molecular species set to be tracked over the KaSim simulation. The CorEx algorithm aggregates measured variables into strongly correlated and dependent subsets. Each group is assigned to one of hidden variables whose number is lower than the set of measured variables. The procedure of finding such partitions is based on optimisation of information theoretic objective that maximises mutual information between grouped variables. In other words, these variable subsets carry most information or best explain each other. At the same time, latent variables become maximally independent from each other.

Latent variables might not specifically refer to any particular meaning but can simplify model structure by grouping measured variables to lower

number of hidden variables that are highly similar to each other. That is, the high dimensional space of measured variables is represented by lower quantity of hidden ones. These latent variables can themselves be divided between higher level and even lower number of hidden variables. In this way, a hierarchical structure is formed providing reduced and abstract representation of the measured data [245].

Strength of dependencies between multiple random variables are measured with *multivariate mutual information*, also called *total correlation* [251]. To outline intuition behind the total correlation, first the *entropy* measure is presented, as a most fundamental measure in the field of information theory. The entropy is a measure of uncertainty in the system or information gain. More formally, entropy H is defined as [252]

$$H(X) = -\sum_{x \in A_x} P(x) \log_2 1/P(x) \tag{3.1}$$

where X is a discrete random variable represented as a set of all its possible individual outcomes, that is $x \in A_x = a_1, ..., a_N$. Each of these outcomes is assigned with a probability $P(x)$. A set of probabilities form a probability distribution over the random variable X, denoted as $P(X)$. The minimal value of entropy, $H(X) = 0$, occurs if $P(X = x) = 1$. It is understood as a complete certainty of the outcome. The maximal value of H is reached when all possible outcomes are equally probable [252].

The entropy definition provides foundations of higher order measures, such as *mutual information*. Mutual information is quantification of dependency between two random variables, X_1 and X_2, defined as follows:

$$\begin{aligned} I(X_1 : X_2) &= H(X_1) + H(X_2) - H(X_1, X_2) \\ &= H(X_1) - H(X_1|X_2) \\ &= H(X_2) - H(X_2|X_1) \\ &= I(X_2 : X_1) \end{aligned} \tag{3.2}$$

In other words, the mutual information is a symmetric measure, $I(X_1 : X_2) = I(X_2 : X_1)$, of how much the entropy of one variable is reduced when the value of the other is known.

Generalisation of mutual information to more than two variables is

called the total correlation. It is defined with the following equation:

$$TC(X_1,...,X_N) = \sum_{i=1}^{N} H(X_i) - H(X_1,...,X_N). \tag{3.3}$$

The total correlation is equal 0 if there is no relation between random variables. Therefore, the entropy of individual variable X_i is equal to the joint entropy of variables occurring together, $H(X_1,...,X_N)$. On the other hand, the total correlation is maximised, if this joint entropy is equal 0 [245].

We can also measure conditional total correlation of the variable X given some variable Y:

$$TC(X_1,...,X_N|Y) = \sum_{i=1}^{N} H(X_i|Y) - H(X_1,...,X_N|Y) \tag{3.4}$$

The conditional total correlation measures to what extend the variable Y explains the total correlation between variables $X_1,...,X_N$.

The main objective of CorEx is to find a minimal number of hidden variables Y, such that the total correlation between all variables (for simplicity $X = X_1,...,X_N$) decreases if conditioned on Y [248]. This principle is expressed by the following equation:

$$TC(X;Y) = TC(X) - TC(X|Y) \tag{3.5}$$

The expression in Equation 3.5 is maximised by the CorEx algorithm. If the conditional total correlation, $TC(X|Y)$, is equal 0, then $TC(X;Y)$ is maximised. It means that Y explains all X. Therefore, an optimal Y minimises $TC(X|Y)$.

CorEx maximises this objective for more than one hidden variable, also called explanatory factors. Each of these factors is a function of a subset of X. For more than one explanatory factors, $Y = Y_1,...,Y_m$, the optimised expression can be defined as

$$\max G_j, P(y_j|x_{G_j}) \sum_{j=1}^{m} TC(X_{G_j};Y_j) \quad s.t. |Y_j| = k \quad G_j \cap G_{j' \neq j} = \emptyset \tag{3.6}$$

where G_j is a subset of indices of all N random variables, $G \subseteq N = 1,...,n$ and X_{G_j} is the subset of the random variables that are grouped over latent variables Y_j. Each of the latent variables can take k discrete values. The algorithm maximises conditional distribution $P(y|x_G)$ [245, 250].

$TC(X_{G_j}; Y_j)$ with respect to some Y_j can reach high values when the dependence between a small number of grouped variables is strong, or if there is a weak dependence between a large number of variables [253]. The number of hidden variables Y is a user-defined input that can be approximately determined with respect to data by probing different values of Y [245].

As CorEx effectively partitions measured variables into clusters, each represented by a hidden variable, the term "cluster" will be used henceforth to denote the term "hidden variable" to simplify naming convention.

The authors use an iterative procedure to efficiently obtain the optimisation of $TC(X; Y)$. Details of applied implementation can be found in Ver Steeg and Galstyan [250] and in Pepke and Ver Steeg [248]. The theoretic method description of the algorithm can be found in Ver Steeg and Galstyan [245, 250]. The algorithm version used here is a most recent implementation of CorEx, "`bio_corex`", that was developed for the study of Pepke and Ver Steeg [248].

3.2.2 Cluster similarity measure

To analyse the impact of different input parameters on clustering results obtained with CorEx, such as the number of clusters that represent the data, similarity between two different CorEx runs is measured by evaluating agreement in members allocation between pairs of clusters.

Comparison between pairs of clusterings are usually aimed to evaluate validity of a clustering method by using *validity indices*. Variable methods defining validity indices are used to compare paired clusterings [246]. These indices are divided into internal and external validity indices. The internal validity indices are based on a pairwise distances matrix derived from the input data. Such properties as tightness and separation of clusters are rewarded [246]. The external validity indices relay on a "ground-truth" clustering, that is a clustering known to be a correct partition of dataset [246]. The external indices are evaluated based on cluster labels. The internal indices additionally require distance metrics between data points as for instance in popular Silhouette Value method for calculating internal indices [254] that is based on between and within cluster distances. To use the internal indices in this context of this study, a sophisticated distance metric that captures complex and non-linear relations existing between trajectories of clustered observables would have to be chosen. It is a rather difficult task that would have to be addressed in a

separate study. Therefore, the aim of cluster comparison is not to evaluate the quality of clusterings but rather establish similarity between clusterings obtained with CorEx. To this purpose ARI was selected as one of the external validity measures that is commonly used for clusterings comparison.

ARI is an adjusted-for-chance version of The Rand Index that us a similarity measure between a pair of clusterings [255]. In non-adjusted measures similarity scores depend heavily on the cluster number to sample number relation, where the score gradually increases with the number of clusters approaching the number of all clustered elements.

ARI can be defined as follows. For a set of elements n and a pair of clusterings, $X = X_1, X_2, ..., X_r$ and $Y = Y_1, Y_2, ..., Y_s$, the overlapping membership can be represented as contingency matrix (*Table 3.2.2*), where n_{ij} is a number of shared elements between X_i and Y_j.

	Y_1	...	Y_s	Sums
X_1	n_{11}	...	n_{1s}	a_1
...	...	...	...	...
X_r	n_{r1}	...	n_{rs}	a_r
Sums	b_1	...	b_s	

Based on the contingency matrix the following equation can be defined:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}] / \binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}] / \binom{n}{2}} \tag{3.7}$$

The equation can be generally represented as:

$$ARI = \frac{Index - ExpectedIndex}{MaximalIndex - ExpectedIndex} \tag{3.8}$$

ARI takes values within a range of 0.0 and 1.0. The index is equal to 1.0 if two clusterings are identical up to a permutation of cluster labels, and 0.0 otherwise. ARI is a symmetric measure that is independent from the ordering of clustering pairs. In this study, computation of ARI was performed with the Python "`scikit-learn`" package where the method is implemented in the function "`adjusted_rand_score`" [256].

3.2.3 Sensitivity analysis for rule-based models

Sensitivity analysis is an quantitative approach to evaluate how the model output is affected by changes in model parameters obtained with calculation of *sensitivity indices*. In the general field of modelling, SA is considered

to be necessary for establishing the credibility of model-based analyses [257]. SA should support model generated predictions used for regulation, policy and decision making [236]. It is so, because "*[..] most simulation models will be complex, with many parameters, state-variables and non-linear relations. Under the best circumstances, such models have many degrees of freedom and, with judicious fiddling, can be made to produce virtually any desired behavior, often with both plausible structure and parameter values*"[258]. This is no different with systems biology models as the study of Gutenkunst et al. [259] showed by analysis of 17 such models. Behaviour of these models were characterised as insensitive to large variations of many parameters, phenomenon called "sloppiness". With such uncertainty in parameter values and difficulty in obtaining direct experimental measurements for all parameter values, attempts to make predictions from such biological models should be supported by methods for refinement and analysis of parameter spectra, such as SA [259].

SA methods are generally divided into two classes, local sensitivity analysis (LSA) and global sensitivity analysis (GSA). LSA was the first one applied to quantitatively evaluate parameter sensitivity scores. LSA involves measuring partial derivative of an output variable with respect to small perturbations around nominal values of an input parameter [260]. Partial derivative is a derivative of a multivariate function that calculates changes in the output of function with respect to variation of one of its variables when the others are kept constant [261]. LSA is the most common method for sensitivity analysis in models of synaptic plasticity [203]. Particular shortcoming of this method is that it assumes model linearity and it only allows small variation of a single parameter at a time [236, 262]. Therefore, LSA might not be informative enough regarding sensitivities of parameters that can have complex and higher order interactions. Small parameter perturbations might not reveal the actual parameter sensitivities, as they need not be aligned with real fluctuations in the biological system [231]. To address these weaknesses GSA was proposed as a new approach. GSA allows for variation of all parameters at the same time across large ranges of values and therefore, enables assessment of sensitivities in a broader parameter space. It has been shown for some models that switching from LSA to GSA can reveal critical parameter values not seen with LSA [231]. It has been demonstrated that GSA can be used to study system robustness to perturbations with a gradual increase in parameter variation levels

[231]. In the context of this study, performing GSA is more preferable as the Fernandez et al. [177] model was already analysed with LSA.

GSA is composed of several steps that involve [263, 264]:

1. specification of the study purpose
2. selection of parameter inputs of interest (whole or subset)
3. specification of variability range of selected inputs (folds or percentage of nominal values)
4. specification of probability distribution for sampling values for selected inputs
5. defining the number of samples
6. application of a sampling method to generate the defined number of samples (step 5) drawn from chosen distribution (step 3) within the predefined range (step 2)
7. evaluation of the model for each sample of parameter inputs
8. estimation of sensitivity indices for each parameter input based on model output results (step 6)

The consideration of the study purpose is an important step as it can determine the choice between different sensitivity indices and their estimation methods. Therefore, it is commonly advised to first define what is the aim of this study [260]. Among many purposes, GSA allows to identify parameters that [236, 265]:

- drive model results by ranking model input parameters according to their scored influence on the model output (*parameter prioritisation*)
- can be fixed to their nominal values during model calibration as they do not contribute to reduction of output uncertainty (*parameter fixing*)
- are most important for given regime of the output value (*parameter mapping*)

Of these three, the most common goal of sensitivity measure is parameter prioritisation. Attainable information about relations between parameters and model outputs can be constrained by the model dimensionality that is the number of parameters involved. A model with higher than a few tens of parameters is usually classified as high-dimensional [260]. It is often discussed that the model size often constrains the method choice and the level of detail obtained from SA. More detailed methods for SA are at the same time more computationally demanding. For instance, compared to LSA, GSA is known to be more computationally expensive but can offer more reliable information on relations existing in the model. For large scale high-dimensional models, the computational cost of GSA can be even prohibitive as the model has to be evaluated a hundreds of times to obtain a sufficient number of samples. However, assumption of linearity of LSA will only offer a partial information on model parameters. To balance the efficiency of SA method and level of detail, it is advised to examine complexity of relations within the model structure, e.g. linearity and monotonicity of input-to-output relations in making decision which SA method to choose [260]. It is known that in most realistic models this assumption cannot be guaranteed. Therefore, selection of SA method is often a trade-off between the amount of information gained from such analysis and the model size.

To approach SA for large scale models, a preliminary screening step is often advised, such as Morris method [265, s. 4.2]. This method can partition model parameters to three groups that differ in characteristics of their effects on the model output: (a) negligible, (b) linear and additive, (c) non-linear or interaction effects indicating that a parameter interacts with others [265, s. 4.2]. In this way parameters with negligible effects can be excluded from SA [260]. As all parameter screening methods, the Morris method provides qualitative ranking of input parameters with respect to their importance but does not quantify the actual difference in importance between parameters [265, p.108]. GSA provides a quantitative measure for importance of parameters and is directly used in this study, without the preliminary screening step.

Research on GSA methods is a vibrant field. Multiple books, reviews and guidelines have been published on GSA methods, for a general modelling domain [236, 262, 265, 266], specific to systems biology [267] and pharmacology [235], offering guidance for good practices and methods comparisons. Here,

I will briefly mention methods in the course of advocating the final choice of an appropriate GSA method for RB models. Particularly applicable to the RB modelling is a prominent study of Marino et al. [268]. The study offers thorough analysis of two most notoriously used, robust and efficient sampling-based methods for GSA, Partial Rank Correlation Coefficient (PRCC) and extended Fourier Amplitude Senstivity Test (eFAST) [269]. Marino et al. [268] proposed a methodology for deterministic and stochastic models based on analysis of variable models from the biological domain. The authors compared results obtained with both GSA methods, and provided methodology to perform and circumvent problems that are common in application of these methods in both deterministic and stochastic frameworks. The methodology presented for the stochastic setting is of particular interest in the context of RB modelling.

Marino et al. [268] indicated two types of uncertainty in stochastic models. The first one, *epistemic uncertainty*, shared with deterministic models, and *aleatory uncertainty*, specific only to stochastic models. The epistemic uncertainty originates from lacking knowledge about the modelled system, whereas the aleatory uncertainty refers to stochastic character of simulated time courses that is their innate feature [268]. To tackle the aleatory uncertainty in stochastic models and to be able to use GSA methods designed for deterministic setting, Marino et al. [268] proposed that simulation of each parameter setting should be repeated multiple times and obtained time courses should be averaged. The study has been frequently used by modelling practitioners as a reference and a guide to use GSA in systems biology [232, 267, 270, 271]. The methodology of Marino et al. [268] has been also used in the context of RB modelling. Sorokin et al. [272] proposed "`RKappa`" package implemented in `R` language, tailored to perform GSA for RB models by using parallelised computation of sensitivity scores on computer clusters. "`RKappa`" is an open source package hosted in the GitHub platform [273]. The package will be used in this study as a ready to use GSA framework for RB models.

Of the two methods for GSA analysed by Marino et al. [268], Sorokin et al. [272] implemented PRCC. The authors noted that the "`RKappa`" framework can be easily extended with other sensitivity methods. In fact, the PRCC implementation used in the "`RKappa`" package comes from the `R` "`sensitivity`" package, devoted to SA and containing a large selection of alternative methods [274].

As a guided use of PRCC and eFAST is presented in Marino et al. [268], these are the first methods to be analysed to decide which of GSA methods is most appropriate for RB model, based on the example of the Kappa model of DARPP-32 presented in *Chapter 2*.

PRCC and eFAST are methods for importance measure [260] that rank model inputs based on their influence on model output. Ranks for each input are calculated with a chosen sensitivity measure. The first to concentrate on is PRCC. It is a highly efficient method that does not require any specific design of experiment or parameter variation. PRCC belongs to a general class of methods relaying on linear regression techniques, such as Pearson Correlation Coefficient (CC), Standard Regression Coefficient (SRC) and Standard Ranked Regression Coefficient (SRRC) [260]. PRCC is Partial Correlation Coefficient (PCC) but applied to ranked transformed data. The ranking step allows to apply this method when output-to-input relationships are non-linear. However, as Marino et al. [268] showed on an example of Lotka-Volterra model [222], PRCC requires monotonicity in input-to-output relations to provide accurate results. To observe if such non-monotonic relations are also present in the Kappa model of DARPP-32, variation of a model observable ("Thr34") was examined against variation in two exemplary parameter values ("`kon41`", "`kcat3`") with scatter plots (FIGURE 3.3), demonstrating that the model has non-monotonic relationships. Therefore, PRCC cannot be used in this model example.

In this case, Marino et al. [268] recommended use of eFAST. eFAST is a model-independent variance-based method, free from assumption on input-to-output relations [269]. Variance is a statistical measure of spread of random variables from their average value. eFAST is an improved variant of FAST, the very first variance-based method, proposed by Cukier et al. [276]. Variance-based methods are a general class of methods that provides sensitivity indices that express how much of variance in the output can be attributed to an input parameter or their combinations [260]. Variance-based indices are believed to be superior measures of uncertainty because of possibility of estimation of the influence of individual parameters or their groups on the model output [277]. Variance-based indices allow to study different orders of interaction effects between parameters by partitioning the output variance into orders of effects, so called partial variances. The *first order*, or *main effect*, is individual contribution

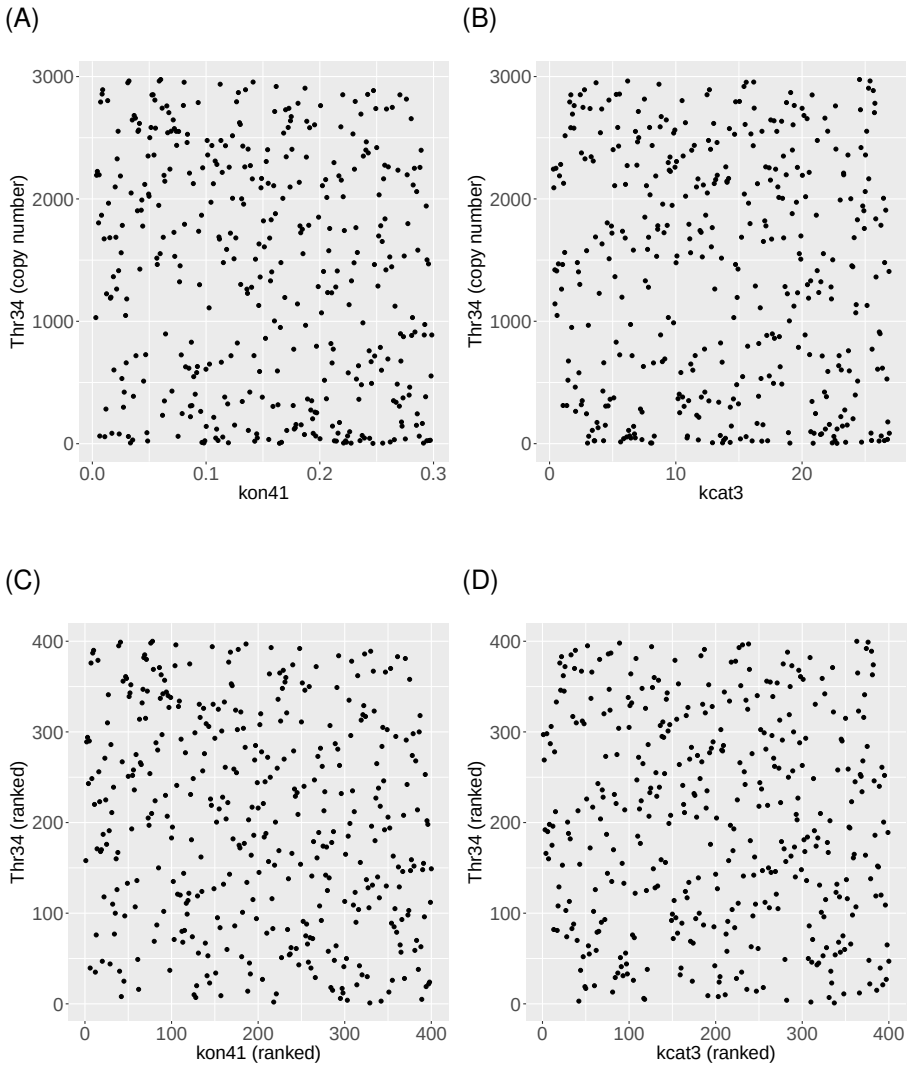


FIGURE 3.3: Complexity of input-to-output relations, such as non-linearities and non-monotonicities, can be assessed with scatter plots of observable-to-parameters [275]. Figures present results of such plots for the "Thr34" observable plotted against two parameters for the 410th second of the simulation. The data was obtained with 400 simulations of 400 varied parameter sets sampled from the Sobol sequence. Copy numbers of the observable trajectories were averaged over 6 repeats of 400 samples (see *Section 3.3* for methodology details). Scatter plots were produced for unprocessed values: (A) "`kon41`", (B) "`kcat3`"; and ranked values (C) "`kon41`" (D) "`kcat3`". According to the reaction set defining the model of Fernandez et al. [177], both parameters are strongly related to the output variable. The first one determines the speed of rebinding "PKA" to "R2C2", therefore removing the kinase from the system, that phosphorylates DARPP-32 at Thr34. The second chosen parameter, "`kcat3`", is a catalytic rate of Thr34 phosphorylation. The results show neither linear, nor monotonic dependence between the input parameters and the output result. It should be noted that the selected timepoint takes place at the start of the cAMP-stimulus that defines the steepest gradient in the simulation. This affected the spread of data points that are shown at its peak.

of a parameter to the output variance. Second and higher order interaction effects indicate contribution of interactions between two and higher number of input parameters, respectively, to the output variance. Interaction effects are interactions between parameters that effects are not the same as addition of their individual first-order effects [265]. Calculation of higher order effects can easily reach a combinatorial barrier with large number of parameters ($2^n - 1$ for n parameters) [269]. To circumvent this issue, *total-effect* indices were proposed as a synergistic sensitivity score that include all interaction sensitivities of a given parameter [278]. For instance, for three parameters X_1, X_2, X_3, if S_{x1} is a first order sensitivity of parameter X_1, then total order sensitivity of this parameter can be represented as $S_{T_1} = S_{x_1} + S_{x_1,x_2} + S_{x_1,x_3} + S_{x_1,x_2,x_3}$ [269]. If $S_{T_1} > S_{x1}$ then the model is non-additive and therefore, interactions exists between $X1$ and other parameters [268]. Existence of interaction terms is very likely if the model has large number of parameters and the model is perturbed within a larger range [269, 279]. In fact the classical FAST method was able to calculate only the first order effect. eFAST, as its extension, was introduced to join two aspects important the detailed decomposition of interactions between inputs and computational efficiency of original FAST method. The eFAST method is able to identify first-order effects and total-effects, as most important effects to interpret sensitivity results [260].

Variance apportioned to a specific input parameter in the output result is identified with Fourier analysis. Fourier analysis decomposes a periodic function into component sinusoidal functions of different frequencies and amplitudes [280]. In eFAST each input variable is varied with a specific frequency that are parameter identities detectable in the output signal with Fourier analysis [269]. How strong parameter frequency is propagated in the model output is a measure of sensitivity [268]. eFAST requires a careful setting of additional parameters relating to this procedure, such as frequencies and the interference parameter. Therefore, the application of eFAST is more complex than PRCC as it requires a specific sampling scheme of parameter inputs and frequency values should be carefully selected to be correctly identified with Fourier analysis.

Application of eFAST to high-dimensional models poses certain drawbacks. It is known to over-estimate sensitivity scores with high-dimensional inputs [268, 281]. Major limitation of applying eFAST is that it has high computational costs when applied to models with a large number of parameters [268].

For instance, analysed by Marino et al. [268] stochastic model with 12 parameters required 53 456 model calls. Despite continuous research on improvement of variance based approaches, calculation of variance-based indices remain a computationally intensive task for high-dimensional input parameters.

Variance summarises the entire probability distribution into a scalar statistics (second-moment), and therefore is an incomplete statistics of uncertainty in the model. As noted by Saltelli [257], if we use variance-based methods for GSA, we "*implicitly assume that this moment is sufficient to describe the output variability*". Variance might not be sufficient to calculate global sensitivity indices if the output distribution is multimodal or strongly skewed [279]. This remark arguments a proposition of moment-independence index, that is *density-based index*, by Borgonovo [279] that takes into account the whole distribution of the output variable. Specifically, the density-based index is defined as calculation of difference between unconditional output distribution, where all parameters are varied, and conditional distribution, when one of the input variables is fixed to a certain value. The author reported that the new type of sensitivity indices can offer a different perspective on model sensitivities than then a variance based method as parameters were differently ranked with respect to their importance when the entire distribution of output was considered compared to the total sensitivity index and the first-order index [279]. However, the approach appeared to be a very computationally intensive task for high-dimensional models that require a specific design for sampling scheme (e.g. double sampling loop) [279]. This lack of computational efficiency locates indices proposed by Borgonovo [279] close to variance-based indices that also have large computational cost. To circumvent this issue, Da Veiga [282] proposed a new class of sensitivity indices based on different variants of probabilistic dependence measures that generalise the density-based index by Borgonovo [279]. These measures are particularly advantageous because they can be efficiently calculated for high-dimensional models. These sensitivity indices allows to use common methods for sampling minimising the complexity of task.

The next section informally introduces the sensitivity index proposed by Da Veiga [282].

Method description The dependence-based sensitivity indices proposed by Da Veiga [282] were developed from implications of the density-based sensitivity index by Borgonovo [279]. Baucells and Borgonovo [283] proposed that the impact of n independent input variables, $X_1,...,X_n$, on the output model, $Y = g(X_1,...,X_n)$ $k = 1,...,n$, can be defined with a function that measures dissimilarity, $d(.)$, between probability distribution of Y (P_Y) and conditional probability distribution of Y given X_k ($P_{Y|X_k}$). $\mathbb{E}_{X_k}$ is defined as expectation with respect to X_k.

$$S_k = \mathbb{E}_{X_k}(d(P_Y, P_{Y|X_k})) \tag{3.9}$$

Da Veiga [282] observed that this dissimilarity measure $d(.)$ can be one of a general family of dissimilarity measures called Csiszár f-divergences of the form:

$$d_f(PY||PY|X^k) = \int f\left(\frac{p_Y(y)}{p_{Y|X^k}(y)}\right) p_{Y|X^k}(y)dy \tag{3.10}$$

where p_Y denote the probability distribution function of Y and $p_{Y|X^k}$ of $Y|X^k$. There are different choices of the function f that denote different measures of distance and divergence, such Kullback–Leibler divergence. Insertion of Equation 3.10 into Equation 3.9 yields a sensitivity index. By virtue of being Csiszár f-divergences these indices are positive and equal 0 if Y and X_k are independent [284]. By substituting different options of function f Da Veiga [282] was able to represent previously proposed sensitivity indices and therefore, enclose them with a general class of Csiszár f-divergences. Da Veiga [282] noticed further that with specific choices of function f transforms the sensitivity index into well known dependence measures, that in general, compare the joint distribution and the product of the marginals. This result opened a possibility to consider other, more recently developed dependence measures and propose new sensitivity indices based on variable dependence measures that can be efficiently computed for multivariate random variables [282].

This study will particularly concentrate on one of these dependence measures, Hilbert-Schmidt Independence Criterion (HSIC). HSIC is a dependence measure based on the Hilbert-Schmidt norm of the cross-covariance operator between Reproducing Kernel Hilbert Spaces (RKHS), introduced by Gretton et al. [285]. It is a sophisticated method that will be briefly outlined her to draw an intuitive understanding underlying HSIC-based indices.

In the domain of machine learning and classification tasks, it is known

that non-linearly separable data sets, if transformed into higher dimensions, can be separated with linear methods. This separation can be performed with a computational efficiency by using methods based on kernel functions. Kernel functions are vary general class of functions that in the field of machine learning are called "similarity" functions. They allow to map inner products between variables to an inner product of these variables in higher dimensions, procedure called as "kernel trick". Generally, the inner product is an important operation that allows to define distances between vectors. HSIC incorporates the kernel trick through RKHS that allows to capture non-linear relationships between inputs in higher dimensional space, like Hilbert space. Hilbert space is mathematically defined space through possible operations, which is a generalisation of Euclidean space to potentially infinite number of dimensions. It allows to use methods of linear algebra and calculus and is equipped with the inner product.

RKHS is a Hilbert space of functions defining relations between functions such that their closeness can be compared. The cross-covariance operator generalises the covariance matrix between X and Y by representing higher order correlations between X and Y through non-linear kernels. In this way, non-linearities and non-monotonicities in the model are tackled [286].

The HSIC sensitivity indices are obtained in a normalised form:

$$S_k^{HSIC} = \frac{HSIC(X_k, Y)}{\sqrt{(HSIC(X_k, X_k)HSIC(Y, Y))}} \tag{3.11}$$

The HSIC method is located in the topic of kernel embedding of distributions in higher dimensions [287]. The formal presentation of the method is provided in the original paper. The method description with more explanatory information regarding Hilbert space and Reproducing Kernel Hilbert Space in the context of Da Veiga [282] work was done by Sinha [284].

Da Veiga [282] compared the sensitivity index based on HSIC to first-order and total indices applied on the Ishigami function. The Ishigami function is non-linear, non-monotonic function with strong interaction term between two parameters that is typically used to compare methods designed to calculate sensitivity scores of models with complex input-to-output relations. Da Veiga [282] observed that HSIC ranked variables with respect to their importance as the total index, when the total index indicated different ranking

from the first-order index. This means that it captures all interaction effects of a parameter but with much lower number of parameter samples necessary to calculate total indices that with the state-of-art variance-based sensitivity methods, such as Random Balance Design FAST (RBD-FAST) [282]. The results also robustly detected non-influential parameters. The HSIC-based indices have been successfully applied and compared in other studies. In the context of modelling molecular pathways, Sinha [284] compared different implementations of variance-based indices (first and total) with the Da Veiga's [282] method on the Wnt signalling pathway in colorectal cancer and reported that the HSIC-based indices are more robust and accurate in indication of drastic differences in parameter importance between normal and tumor representing models. The HSIC-based method with adjustments was also used for sensitivity analysis of large spatio-temporal models of radionuclide concentrations [288] and for parameter screening [286].

3.3 Methodology

The aim of this chapter is to propose a pipeline for prioritisation of observables and parameters of RB model that application is presented on an example of the RB model analysed in *Chapter 2*. The following sections will chronologically discuss each step of the pipeline that incorporate the methods described in the previous section. Datasets analysed in the pipeline are time traces of observables obtained with simulations of the RB model. These time traces are analysed with CorEx that identifies subsets of most dependent observables and possibly prioritise one of them as most intertwined. With respect to chosen subset(s) important parameters are identified with the HSIC-based GSA. Metrics obtained with CorEx and HSIC are integrated as weighted network graphs.

In the first part of this section, I will outline the pipeline scheme. In the next part, the choice of model observables that are input variables to the CorEx step is discussed. The number of prioritised observables is reduced to a subset when passed to the GSA step. At the final step, a network is constructed integrating information from two previous steps. The following section will also present and discuss method-specific settings.

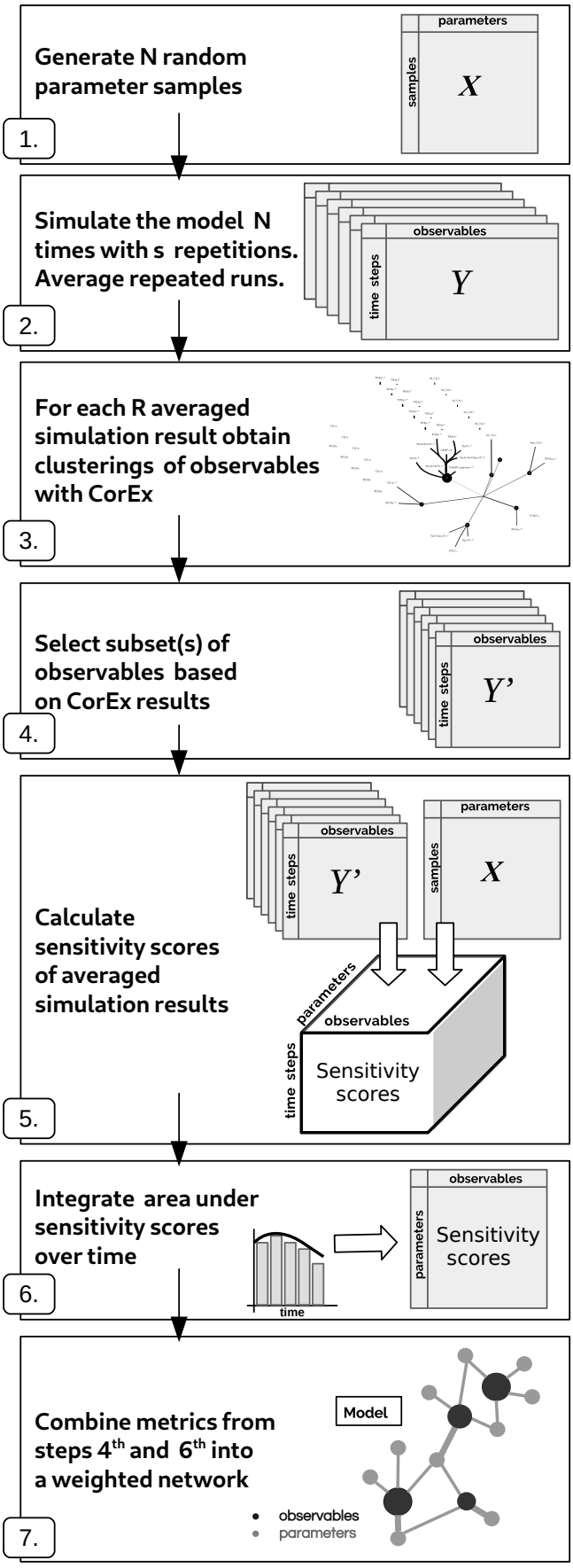


FIGURE 3.4: Overview of the pipeline steps. See main text for details.

3.3.1 Pipeline overview

FIGURE 3.4 outlines a general skeleton of the pipeline. The pipeline operates on the same model results in both CorEx (step 3) and GSA (step 5). Therefore, from the very first step the procedure requires to conform to requirements of GSA. As the HSIC-based method for GSA does not require any specific design to obtain parameter samples, the steps are reduced to general stages of GSA. In the first stage, given nominal values of all rate constants, called parameters X, a random vector N of parameter inputs is generated with a selected sampling method over a chosen probability distribution and drawn from specified range. In the next step, the model is simulated with each parameter set N to obtain time-courses of observables as outputs Y for each sample. The simulation of each parameter set is repeated s times and averaged over repeated runs. These steps are already encoded in "`RKappa`" package [272] that is used as a encompassing framework for performing GSA on RB models in this study.

In the next (step 3), the averaged simulation results are passed to CorEx to generate N different clusterings of observables. Based on aggregated score of multiple metrics provided by CorEx, the 4th step involves selecting subgroups of observables. In the 5th step, sensitivity scores are calculated between X and a selected subset of observables Y'. In the 6th step, a three-dimensional matrix associating sensitivity score to each parameter per time per observable is reduced to two dimensions by integrating change of sensitivity score over time. In the last 7th step, scores of parameters and observables are integrated into a weighted network representing the model in a reduced form. This integrated representation allows to analyse the model in a summarised form with methods derived from graph theory.

The following sections present each of these steps in detail combined with analysis and discussion on parameter settings used in each method. First, however, I present different sets of observables selected to test the pipeline with. The first observable set is aligned with the ones defined by the authors of the ODE model, Fernandez et al. [177]. The second set is composed of automatically generated observables obtained during the simulation of the RB model with snapshots of molecular mixture.

3.3.2 Observable sets

The pipeline is applied to time-courses obtained with the RB model of DARPP-32 network in the competitive variant of the model where DARPP-32 has one binding site (*Section 2.3.2.1*). Two types of model readouts were selected to be tracked over the KaSim simulation. The first one consists of 19 observables. 12 of these observables were defined as the first 12 in *Table 2.1* (*Chapter 2*). The remaining 7 observables correspond to the last 3 in *Table 2.1* ("D34", "D75" and "D137") with the difference that DARPP-32 species that are analysed in this chapter, are represented with explicit enumeration of all possible combinations of phosphorylation states on three sites. In this way, selected observables form disjoint sets of molecular species. For instance, molecular species composing the observable "D34" in *Table 2.1* are explicitly represented by four observables in the observable set analysed in this chapter, that is: "Thr34", "Thr34:Thr75", "Thr34:Ser137" and "Thr34:Thr75:Ser137". In all these 19 observable expressions, the binding state of DARPP-32 is unspecified.

The other set of variables is retrieved in an automated manner with the `$SNAPSHOT` command (see *Section 1.5.1.5* for the snapshot definition), used to record molecular species over the model simulation by taking snapshots of current states of molecular mixture. The procedure of recording snapshots was defined in *Section 2.4.1* and encoded with Code 2.10, where the snapshot was taken every 10000th productive event (`[E+]`). Application of this procedure resulted in 9322 snapshots, that are parsed to retrieve a unique set of expressions defining 91 molecular species. These expressions were transformed into an input file that determines tracked observables during the model simulation.

3.3.3 Setup of CorEx input parameters

There are a few user-specified input parameters in CorEx that particular choice should be elaborated with discussion. These are: (1) number of clusters Y, (2) number of discrete states or dimensions of clusters (the same for each Y_j), (3) number of iterations during which $TC(X;Y_j)$ is optimised, and (4) number of automated repetitions of CorEx runs. Among these are parameters that determination require exploration of parameter ranges with multiple CorEx runs (1,3), and parameters that determination can relay on interpretation of previous studies in the context of this study (2,4). Except for the number of clusters, setup of the other three parameters is discussed in this section.

Outcomes of explorations of the number of clusters are presented and discussed in the result section (*Section 3.4.1*) as they reveal important characteristics of CorEx results with respect to the datasets. This exploratory approach was supported with a convergence measure provided as a standard output of CorEx execution. The convergence measure informs if over the course of iterations CorEx successfully learned a fixed solution that does not oscillate between disparate values at the final iterations. The convergence measure is based on the sum of cluster strengths, $\sum TC(X;Y_j)$, calculated for each iteration. To calculate this measure, $\sum TC(X;Y_j)$ from the last ten iterations are divided into two groups of five and the absolute difference between their mean values define the convergence measure. If the difference is smaller than some predefined ϵ, by default set to $1e-05$, then the algorithm converged.

The number of cluster dimensions was fixed to the default value of 2. This particular values has an intuitive interpretation of Y_j being either "high" or "low" [253]. In all but one published study applying CorEx, the value for the cluster dimensions was set to default [245, 249, 250]. The exception was the study of gene expression dataset by Pepke and Ver Steeg [248], where dimensions were set to 3, that is the highest number of dimensions that CorEx was tested with [253]. This particular number of dimensions was chosen because of its straightforward translation into levels of gene expression: "under", "neutral" and "over" expressed. These 3 labels were then used to divide expression samples within gene clusters. As in this study there is no particular number of bins or labels that clusters could be stratified with and convey biological meaning; therefore, this value was left as default. It is worth noting that increase of dimensions results with increase in computational cost. In the preliminary screenings in this study, CorEx was executed with a set of higher dimensions then default. It was observed that with higher dimensions, CorEx is less likely to converge (data not shown). This additionally tips the balance against using higher dimensions of Y_j in this study.

Supported with examination of multiple CorEx runs, the number of iterations is set to 500. It is 5 times the default number. Increase of the number of iterations above 500 did not improve cases when CorEx runs resulted in sustained and distinctive oscillations between $\sum TC(X;Y_j)$ values indicating impossibility of convergence. The setup of 500 iterations is a very safe margin as learning process converges much earlier. This precociousness is dictated by

uncertainty of perturbed model results.

Because CorEx is only guaranteed to arrive at local optimum, the CorEx implementation provides a user-specified input for automated repetition of CorEx runs that returns results of the largest $\sum TC(X;Y_j)$. For this study, this number is set to 3.

A particular asset of CorEx that was not put to use in this study is that CorEx can learn multi-layered hierarchy of clusters, such that the data set can be represented with gradually smaller number of clusters. In this study, only a single layered clusters were learned as results of two layers produced one significantly distinguished and strongest cluster that was disconnected from others. Moreover, the multi-layered hierarchy of clusters would not serve the purpose of this study.

3.3.4 Parameter sampling and model simulations

This section details first two steps of the pipeline that involve simulation settings and data preprocessing to generate sets of random parameter samples (step 1) and performing model simulations with these randomised parameter sets (step 2). The implementation of both steps can be found in the "`RKappa`" implemented in `R` language, and is used in this study. The package automatically prepares the setup for GSA by generating N separate model specifications each containing one of N sets of generated random parameter samples. Each model is run multiple times according to the predefined number of replicates per parameter sample. Simulations are performed with the KaSim simulator. The biological time of simulation is set to 600 seconds, registering two time points per second that results with time courses of 1200 steps.

As calculation of HSIC-based indices does not require any specific sampling scheme, a sampling method predefined in "`RKappa`" is applied. Parameter sets are sampled with a Quasi-Monte Carlo low-discrepancy sampling, Sobol sequence, with addition of scrambling (randomisation) method of Owen, that preserves the low-discrepancy between samples [289]. This sampling method is implemented in the "`randtoolbox`" package [290]. Parameters sampled with Sobol sequence and Owen's scrambling result with a uniformly distributed samples (Figure 3.5).

Parameter values are sampled within predefined ranges. To encompass a large parameter space the parameters are varied within ±10 folds of their

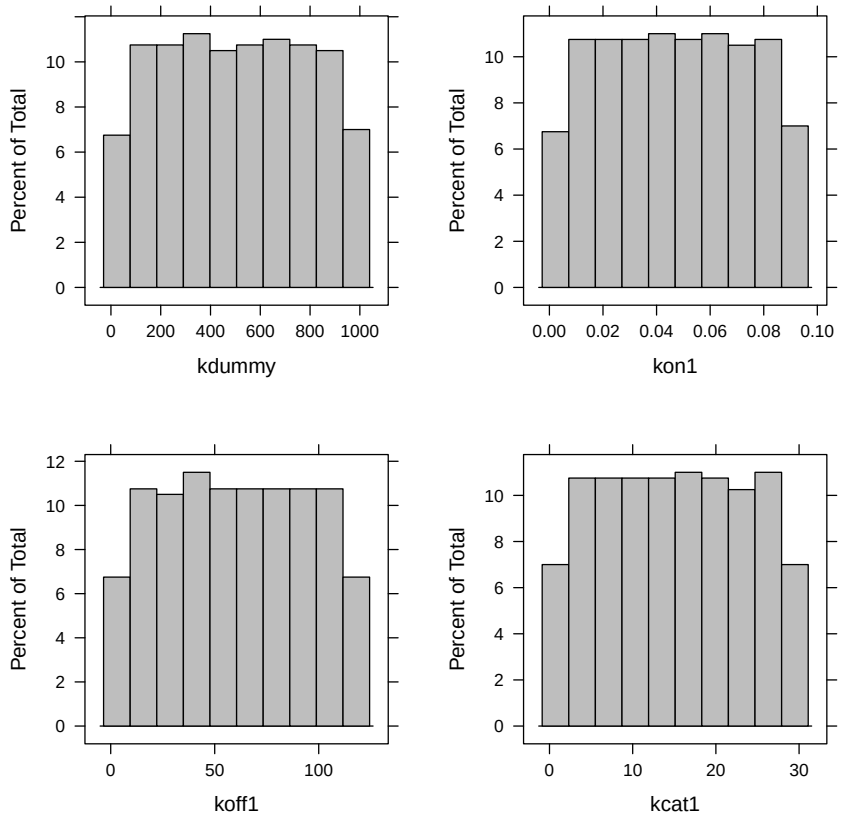


FIGURE 3.5: Distribution of parameter samples of four parameters obtained with the Sobol sequence and the scrambling method of Owen. Distributions of parameter samples are evenly dispersed with less frequently sampled values from both extremes of indicated parameter ranges.

baseline values. Though, the model of Fernandez et al. [177] has not been tested for robustness with any of GSA methods, it has been observed in previews studies that models of the DARPP-32 network are very robust [177, 201] and therefore, stronger parameter variations might create a more appropriate setting to expose important parameter relations. Moreover, variation of parameters in higher ranges reveals prominent importance of interaction effects that become more dominating than first-order effects [269, 279]. Simulations were run in parallel on the University of Edinburgh's computer cluster, Eddie Mark 3. The cluster uses the Open Grid Scheduler that is a batch-queuing system on the Scientific Linux 7 operating system. The batch-queuing system controls scheduling of unattended execution of programs run on a computer cluster [291].

To evaluate a minimal number of required parameter samples, Marino et al. [268] suggested Top-Down Coefficient of Concordance (TDCC). TDCC measures agreement between two sets of ranked parameters emphasising agreement between the top ranks. This sensitivity to agreement on the top ranks is achieved by transforming parameter rankings with Savage scores [292]. This is particularly advantageous in the context of SA as the top sensitivity scores indicate most influential parameters [268, 292]. Similarity between a pair of transformed ranks is measured with correlation that scores one when the top ranks are in complete agreement and zero otherwise. By using TDCC, we can evaluate difference between two parameter rankings obtained with varied numbers of parameter samples. The first step of TDCC requires to define a maximally large number of parameter sets that the model is simulated with. Subsequently, sensitivity indices are computed from time courses and parameter samples divided into subsets with gradually incrementing number of included parameter samples in each consecutive subset. Obtained parameter sensitivity scores, separately for different sample groups, are ranked and transformed with Savage scores to calculate correlation between pairs of subsequent subsets with gradually lower number of samples. If there is no distinctive change between consecutive scores, the increase in number of samples has no effect on the GSA scores.

The implementation of TDCC contained in the "`RKappa`" package is used to evaluate decision on the final number of parameter samples. Figure 3.6 presents results of application of TDCC to the RB model of the DARPP-32 net-

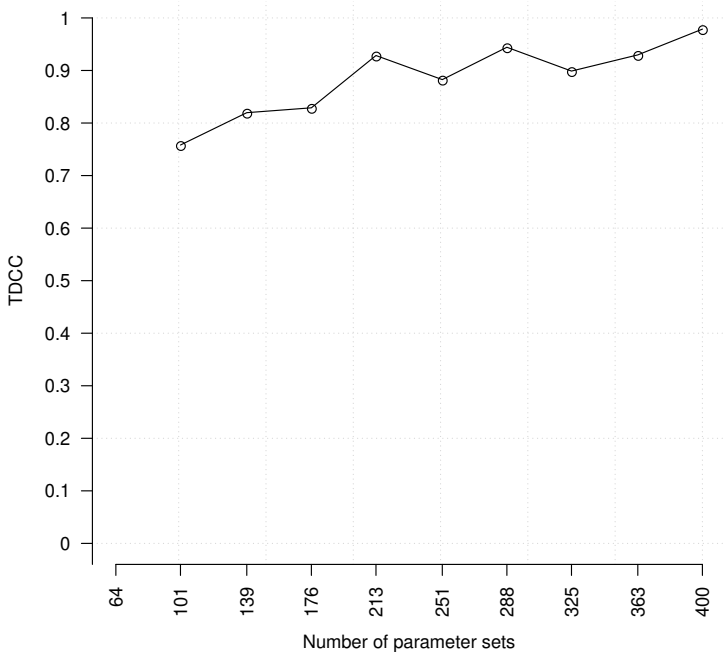


FIGURE 3.6: TDCC scores calculated between pairs of ranked HSIC sensitivity indices calculated for ten subsets of gradually incrementing parameter samples. With relatively low error, a convergence to an agreement between consecutive ranked sensitivity indices can be observed. Based on these results, it can be assumed that 200 parameter samples is sufficient to calculate HSIC-based sensitivity indices for a model with 60 parameters.

work, calculated for the time-course of the "Thr34" observable. The top number of parameter samples was set to 400 and parameter sensitivities were calculated with HSIC. Results are divided into ten subsets, each with incrementing number of samples. TDCC indicates that there is an acceptable level of agreement between scores above the subset with 200 parameter samples. Despite this result, the final number of parameter samples was set to 400 as computational burden implicated in performing model simulations was lowered by parallelising model runs on the computer cluster.

As advised by Marino et al. [268], time courses obtained with each parameter sample should be averaged over multiple model runs to remove the alleatory uncertainty. The number of such repeated runs should be kept above 3. However, automated batch simulations run on computer clusters can be corrupted either due to an error related to the computer cluster or the KaSim simulator, what results with missing data sets. Although CorEx method is indifferent to missing data points, GSA step requires complete data

sets. Therefore, measures to avert incomplete datasets were undertaken.

Simulation results can be affected in two ways. The KaSim simulation can be interrupted that results with shortened time-series, or a lower number of simulation repetitions per a parameter sample is executed than indicated. One of the contributing factors to such interruptions is insufficient amount of virtual memory allocated to a batch job submitted to the Open Grid Scheduler. This can be easily avoided by rising job memory allocation above the default level. In this study, the limit of virtual memory was set to 32G. In spite of testing even higher limits, around 5% of simulations were still interrupted. Therefore, to design simulations against such failures, the size of parameter sets and the number of repeated model runs per parameter set has to be higher than suggested minimal of three. This increased number of repetitions of model runs can defend against loosing any parameter samples that would require re-simulations of lost samples. Therefore, the number of repetitions is set to 6. The same number of repetitions was applied in calculation of TDCC. To indicate the level of abnormalities in the data, simulation results are subjected to tests and editing. In the first step, time courses shorter than 600 seconds are detected and removed. In the following step, replicates of model runs per parameter sample are counted to remove these ones that have ≤3 replicas. If the simulation results show missing samples due to such removal, the model is simulated with the lost ones. However, this problem did not occur as the number of lost sample replicates was never higher than 1.

3.3.5 Selecting subsets of observables with CorEx

In this section, the 4th step of the pipeline is discussed that involve grouping and prioritisation of observables based on analysis of times courses with CorEx. Observables are selected with specifically developed scores for this purpose. These scores decide which observables are progressed to the 5th step of the pipeline, that quantifies the importance of parameters for selected observables. These observable scores are defined through composite metrics. First, I separately characterise each constituent metric, and based on these descriptions, two variants of combined observable scores are proposed.

The constituent metrics are quantities resulting from the CorEx application on averaged time courses obtained with all sampled sets of parameters. These time courses are used to partition observables into groups, called *clus-*

terings. The way observables are partitioned might reoccur despite parameter variations. These recurring clusterings can define a clustering type. We can assume that a clustering type that is more frequent then the others, is more emblematic for the modelled system. Therefore, frequency of clustering types is an important metric to examine. First, we have to ask if there are repeating clusterings. If yes, then how many times a clustering of a certain type appears, i.e. the frequency of clustering type. Clustering frequency can be detected by calculating Adjusted Rand Index (ARI) (*Section 3.2.2*) for each pair of clusterings. Only pairs of clusterings that scored exactly 1 are taken into account. This implies that a given clustering has to appear at least twice to obtain the score. In the next step, all clustering pairs are divided into groups of the same type. Assigning multiple clusterings to the same type is performed based on a simple rule as follows: if clusterings $c1$ and $c2$ scored 1 with ARI, and therefore are the same, and the same applies to clustering $c2$ and $c3$, then all three clusterings are declared as members of the same clustering type, C_k, defined by its constituent members: $c1$, $c2$, $c3$. The frequency ratio calculated for each cluster type is defined as follows:

$$F_k = \frac{|C_k|}{N} \tag{3.12}$$

where N is the parameter sample size, and $|C_k|$ is the cardinality of the clustering set of type k.

CorEx reports strength of every cluster j within each clustering. This cluster characterising metric is the total correlation score (TC_j) defined in Equation 3.5 and further called $TC_j = TC(X; Y_j)$. The cluster strength defines strength of dependence between cluster members.

For each cluster member, CorEx gives also an observable characterising metric, defined through its relation with the cluster it was assigned to. This observable characteristic metric quantifying strength of dependence between observable and the cluster it was assigned to, is defined with mutual information, $MIS_{ji} = I(X_i : Y_j)$, where j is a cluster index and i is an observable index. TC_j and MIS_{ji} define the cluster strength and the observable strength, respectively.

I propose two ways of understanding the observable score. The first one is based on a premise that an observable is important within the group context and its importance might change with variation of rate constants. Therefore, the first score of observable is defined with respect to identified clustering

types.

Definition of observable score with respect to the clustering of type k is as follows:

$$ObsSc_{k,j,i} = F_k + med(TC'_{k,j}) + med(MIS_{k,j,i}) \tag{3.13}$$

F_k is a frequency of clustering type k. For all clustering type members, $med(TC'_{k,j})$ is median of normalised values of cluster strengths for each cluster j, and $med(MIS_{k,j,i})$ is median of observable strengths for each observable i. The median was chosen as a summarising statistics over all clustering type members to make the score robust to outliers. As CorEx finds local optimum, some of its executions might produce such outliers, what may heavily influence the observable scores. TC'_j denotes a normalised value of TC_j. Normalisation is performed with Equation 3.14.

$$TC'_j = \frac{TC_j}{m} \tag{3.14}$$

The variable m is a number of cluster members that is very close to the theoretically maximal cluster score. The maximal value of cluster strength is defined as $m-1$ and it is valid only if dimensions of clusters Y_j are equal 2 [253]. The maximal value of cluster strength was not used in the normalisation equation as the two element clusters then tend to be over-scored. The normalisation of TC_j with Equation 3.14 promotes clusters with high TC scores, relative to the number of members. If the cluster member count is equal 1, TC'_j is set to 0. This means that one element-clusters are understood as degenerate ones. To equally treat the cluster and observable strengths, all MIS values for one-element clusters are also set to 0.

To select the subgroup of observables, first the most frequent clustering type is determined. To determine such clustering type, a matrix of Obs_{kji} scores (*Table 3.1*) is summed over all observables. As all constitutive metrics take values between 0 and 1 and each observable is classified to only one cluster per clustering type, F_k value exposes such clustering type. Then, observables are chosen that belong to the maximally scored cluster that belongs to this highly scored clustering type. The observables selected based on this metric are members of strongest and most frequent clusters. If there would be more than one strong cluster, then each subgroup of observables could be separately progressed to the following pipeline steps. Here, only the strongest cluster is considered.

	$C1$		$C2$	
	$G0$	$G1$	$G0$	$G1$
$O1$	$ObsSc_{C1,G0,01}$	0	$ObsSc_{C2,G0,01}$	0
$O2$	0	$ObsSc_{C1,G1,02}$	0	$ObsSc_{C2,G1,02}$
$O3$	$ObsSc_{C1,G0,03}$	0	$ObsSc_{C2,G0,03}$	0

TABLE 3.1: Matrix of observable scores obtained with application of Equation 3.13. Each observable score ($ObsSc_{k,j,i}$) is composed of a frequency score of clustering type (C_k) and cluster strength it was classified to (Gj) and observable strength defined by relation of the observable (Oi) to the cluster (Gj).

By removing the term of the observable strength in Equation 3.13, we can also define a cluster score as follows:

$$CluSc_{k,j} = F_k + med(TC'_{k,j}) \tag{3.15}$$

The second observable score is defined with Equation 3.16 as a more generic term that include output measures derived from CorEx for all clusterings of parameter samples without differentiating them into types.

$$ObsSc_{ji} = med(TC'_{ji}) + med(MIS_{ji}) \tag{3.16}$$

This alternative definition is prompted by the assumption that despite parameter perturbations, there might be a set of observables that importance is persistent and therefore, these observables are always associated to the most strongly dependent cluster. This score might be also preferred in case no clustering type could be identified. Equation 3.16 is composed of two terms: $med(TC'_{ji})$ is a median value of all normalised strengths of clusters (TC'_j) that the observable i was associated to, and $med(MIS_{ji})$ is a median value of observable strengths (MIS_{ji}) with respect to these clusters j across all clustering types. The normalisation of TC'_i is obtained as in the first observable score, with Equation 3.14.

3.3.6 Calculating and integrating sensitivity scores

In the 5th pipeline step, the parameter samples and the simulation results obtained in steps 1th and 2nd (*Section 3.3.4*), are used to calculate sensitivity scores with the HSIC-based sensitivity indices for selected observables

in the 4th step. In this study timed-GSA is applied [268]. The timed-GSA involves calculation of sensitivity indices for each time step per observable per parameter. In usual practice, a single or at most a handful of time points is chosen to perform GSA [231, 268]. Here, GSA is applied to a time slice from the 402th time step, when the cyclic adenosine monophosphate (cAMP) pulse is introduced, to the 1200th time step, that is the end of simulation. The choice of this particular time-slice for evaluating sensitivity scores is dictated by the fact that the steady state and the stimuli introduction are two different states of the system that might confuse sensitivity profiles. Separation and selection of one of the two system states might clarify the biological interpretation.

Based on sensitivity scores learned in the 5th step, in the 6th step, the timed-GSA results are summarised by taking integral of the area under sensitivity scores along the time axis. Integration was performed with the composite trapezoidal rule. This is a technique for approximating the definite integral of a region under the curve with trapezoids and calculating their area. The parameter defining spacing between sample points was set to 0.5. The composite trapezoidal rule is implemented in Python language in the "`scipy`" package as the "`numpy.trapz`" function [293].

3.3.7 Score consolidation: weighted network of observables and parameters

The scores learned in steps 4th and 6th are unified and represented as a network graph in the 7th pipeline step. The network graph is composed of two kinds of nodes, subset of observables and all parameters. For this reason, network edges join two different kinds of nodes forming a bipartite graph. Parameter sensitivity scores are represented as edge weights. As GSA is performed on more than one model output, network representation can facilitate examination of relations between groups of parameters and observables. With such network representation, results can be viewed with different stringencies on parameters from least to most sensitive with respect to groups of observables by applying cut-off values to edge weights. Moreover, we can compare two varied conditions of the same model by analysis of two network structures that represent these conditions. Because all non-zero parameters are included in the network, the main focus of such analysis is placed on difference between edge weights dependent on the condition. As this study is performed on a

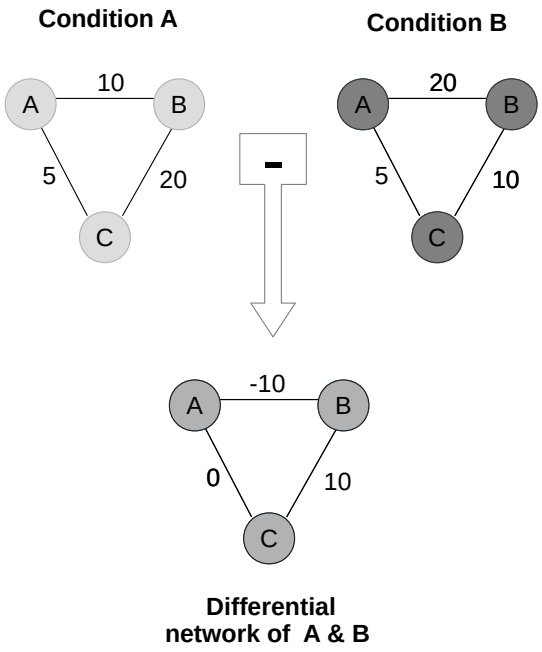


FIGURE 3.7: Approach to analysis of differences between networks representing two different model conditions. Each networks is constructed from observables and parameters connected with edges weighted by sensitivity scores. By subtracting edge weights of a network representing condition “A” from a networks of condition “B”, a new differential network is created that has positive edge weights for these relations that gained importance in the condition “A”, negative edge weights for these that lost importance, and zero-edge weights for unchanged relations.

relatively small network, to observe such differences we can resort to a simple method based on taking the difference in edge weights between a pair of networks representing two different conditions. The variability between conditions is then defined by the gain or loss of edge weights. The 4th pipeline step, where subsets of observables are scored and selected, different observable sets can be excluded from the GSA step dependent on the condition. Therefore, observable nodes can vary between these networks. To be able to compare such networks that include different observables, first we need to reconstruct a complete network containing all observables and assure that they all are connected with each parameter. This is done by adding missing observables to the list of nodes. Subsequently, network edges are drawn between missing observables and all parameters with zero weights. Having two complete networks for two different model conditions, we can subtract edge weights in the first graph from the ones in the second graph. In this way, we obtain a new network, that has edges with positive, negative and zero weights. The positive edge weight denotes that the relation between joint nodes gained importance in the first condition. The negative edge weight means that this relation lost importance in the first condition. The zero-edge means that there is no change between conditions regarding joint nodes (FIGURE 3.7). All three conditions can be visualised as separate networks.

The network view on relations between observables and parameters allows to potentially shift the focus from liner rankings of most dominant model elements, either parameters or observables, to networks of shared relations between these elements. Moreover, by constructing a differential networks between two conditions, we can ask how these relations change due to change in conditions [294].

3.4 Results

Now that we learned details of the pipeline steps, we can progress to presentation of results. The pipeline is applied to the base-line variant ("wild-type") of the DARPP-32 network model and one of two site-directed mutations, the constitutive Ser137 (constSer137), presented in *Chapter 2*.

The pipeline is composed of two distinctive methods, CorEx and HSIC, and divides into five distinctive stages: (1) clustering of time courses with CorEx, (2) scoring of observables with developed metric, (3) parameter scoring

with HSIC-based sensitivity indices, (4) analysis of combined metrics with a network graph, and (5) comparison of two networks representing different model conditions. Results obtained with these five stages require presentation and analyses in separation.

First, I examine CorEx by running the algorithm with time courses for each of two observable sets presented in *Section 3.3.2*. The first set was composed of 19 observables and the other of 91 observables automatically derived from snapshots of molecular mixtures taken during the simulation. Application of CorEx requires decision on the number of clusters. Therefore, first analysis of multiple CorEx runs with varied numbers of clusters for the two observable sets is presented, to decide which one is most appropriate to each set. When the number of clusters is decided, partitions of time courses of the two observable sets obtained with CorEx are compared and analysed.

Following CorEx centred analyses, results of two types of observable scores are shown for the 19-observable time-courses of the "wild-type" model condition. Each composite metric of observable scores is separately demonstrated. Designated subset of observables is progressed to GSA, where integrals of HSIC-based indices that quantify observable-to-parameter relations are analysed. These are then used to construct a weighted network of observables and parameters. The same procedure is applied to the constSer137 model condition. Networks representing two different conditions, the "wild-type" and constSer137, are joint into a "differential network" by subtracting edge weights, on which network analyses are performed.

3.4.1 Determining number of clusters and characterising multiple CorEx runs

Among CorEx input parameters are ones that determination require analysis of multiple CorEx runs with varied values. Such exploratory approach is applied to find the appropriate range of the number of clusters. This value is separately established for the 19- and 91-observable sets. Multiple CorEx runs were performed on time courses obtained from a single model simulation without averaging.

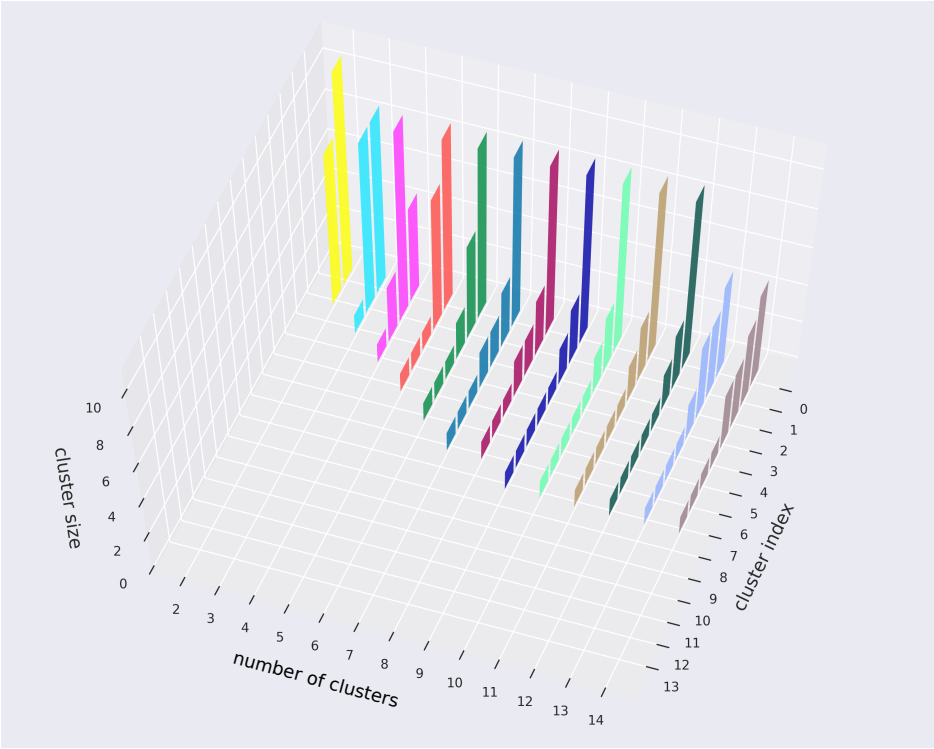
First, the number of clusters is determined for time courses of the 19-observable set. The number of clusters was varied from 2 to 14, each with the default number of dimensions. With these preliminary CorEx runs, alongside

the determination of this input parameter, we can also learn about some general characteristics of clustering results in application to this particular dataset.

FIGURES 3.8 reports results of variations in the number of clusters with respect to three cluster characteristic values: cluster size (FIGURE 3.8A), cluster strength (FIGURE 3.8B), and the normalised cluster strength with Equation 3.14, defined as a ratio between the cluster strength and the cluster size (FIGURE 3.8C). FIGURE 3.8A demonstrates that starting from 9 clusters , 8 of them are assigned with at least one observable and this number remains constant up to the largest number of clusters. The size of the largest cluster, located on 0th index, stays of the constant cluster size of 8 between 3 to 12 clusters , except for 4 clusters. FIGURE 3.8B shows that the drop in counts of members in the first cluster of the clustering with 4 clusters is reflected in its strength, similarly to clusterings with 13 and 14 clusters. It is worth noting that through all values of cluster counts there is a distinctive discrepancy in the cluster size and strength between 0th index cluster and the other clusters. This difference is preserved despite the normalisation (FIGURE 3.8C). In general, clusters with higher than 0th index drop to low values of the cluster strength, closely approaching 0, that is particularly the case in one-member clusters. This clear domination of one cluster indicates that there is a single cluster of observables that will be prioritised to the next pipeline steps.

Knowing that such cluster characteristics as the strength and the member count stabilise with the increase of the number of clusters, we can examine agreement in member allocation between clusterings with earlier introduced Adjusted Rand Index (ARI) (*Section 3.2.2*). FIGURES 3.9 show results of these examination of paired clusterings as they were originally divided by CorEx (FIGURES 3.9A), and transformed clusterings, each composed of two clusters, where the first one consists of observables associated to 0th index cluster, and the second is composed from members of all other clusters (FIGURES 3.9B). Application of ARI to transformed clusterings is in fact a test of agreement on the content of the largest and the strongest cluster with respect to change in the number of clusters. The adjustment for chance of ARI guarantees independence between the number of clusters and the number of samples. In non-adjusted measures, the more number of clusters is closer to the number of samples, the higher is the score of agreement between clusterings [295]. The transformation of clusterings is introduced because the largest cluster towers

(A)



(B)

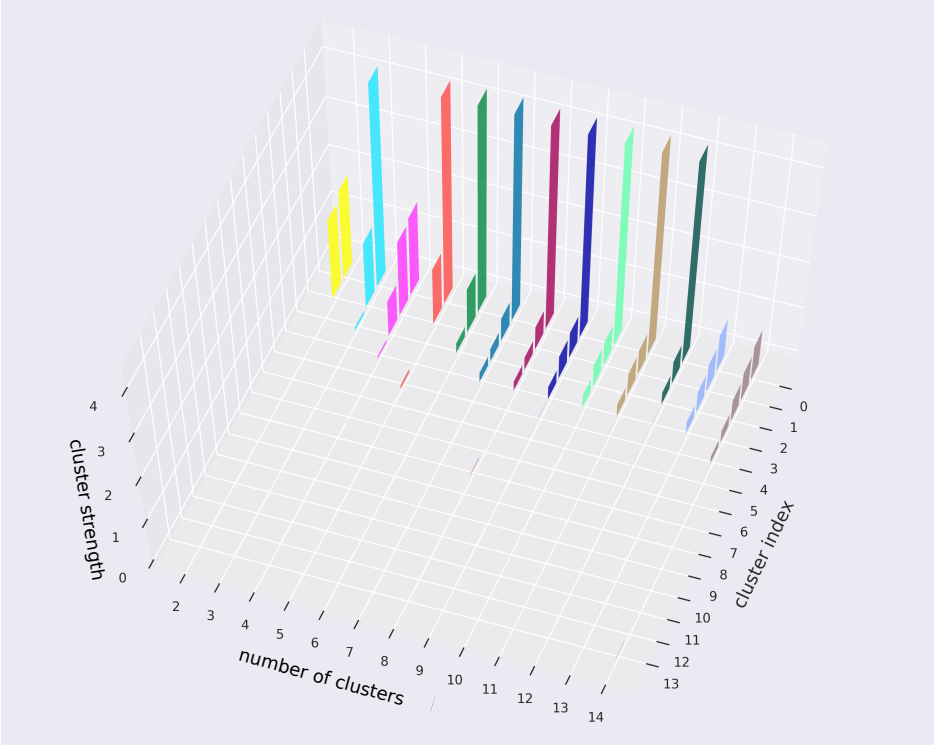


FIGURE 3.8: To determine the most suitable number of clusters for the 19-observable set, clusterings with variable numbers of clusters were obtained. Change in two characteristic values of clusters are examined with respect to these variations: cluster size (A), cluster strength (B). Despite increase in cluster counts to up to 14, maximally 8 clusters have at least one associated cluster member. Distinctive discrepancy between the 0th-indexed cluster and others indicates existence of a single dominating one.

(C)

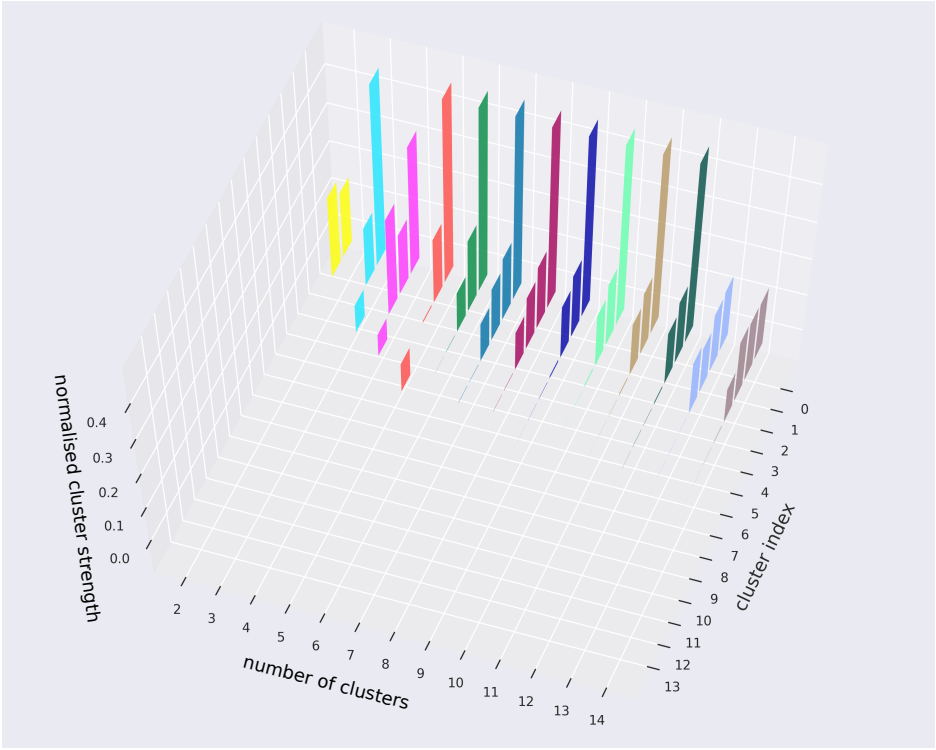


FIGURE 3.8: Cluster characteristic metrics shown in (A) & (B) are combined into a normalised form of cluster strength, that is a ratio between cluster strength and cluster size, and plotted against variation in the number of clusters (C). This ratio is a term comprising the observable scores presented in *Section 3.3.5*, with the difference that here the cluster strength of one element clusters were not set to 0. Examples of 3 to 5 clusters demonstrate that the cluster strength can have negative values. Negative cluster strength is treated in the same way as the 0-cluster strength, and understood as a failure of finding correlation between variables [253]. The ratio exposes the fact that there are multiple observables associated to clusters with negative strength.

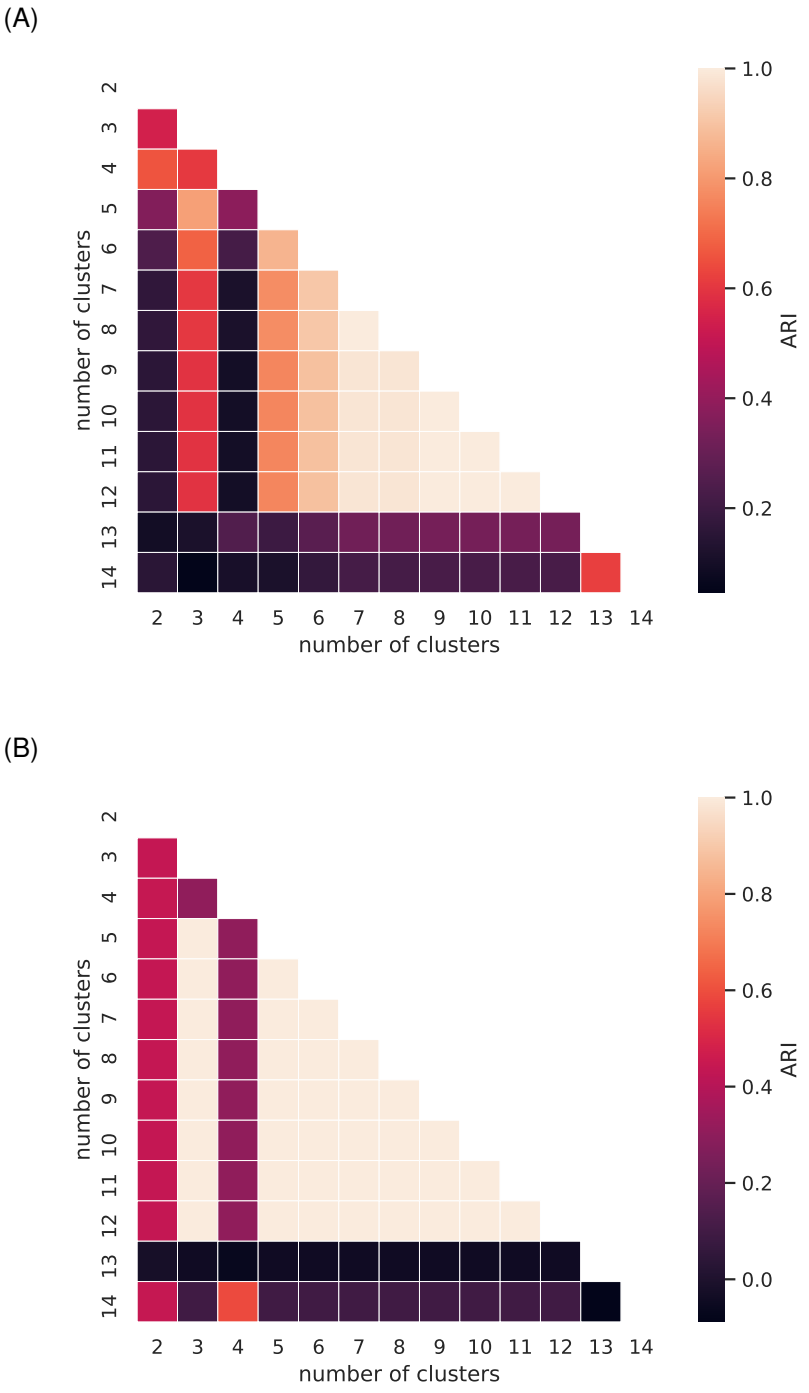


FIGURE 3.9: Agreement in observable allocation between clusterings obtained for different and increasing numbers of clusters computed for time courses of the 19-observable set. The agreement between pairs of clusterings was calculated with ARI, for original clusterings (A), and transformed clusterings (B), composed of two clusters, where the first cluster has members that belong to the largest and the strongest cluster, and the second cluster contains members from all remaining clusters.

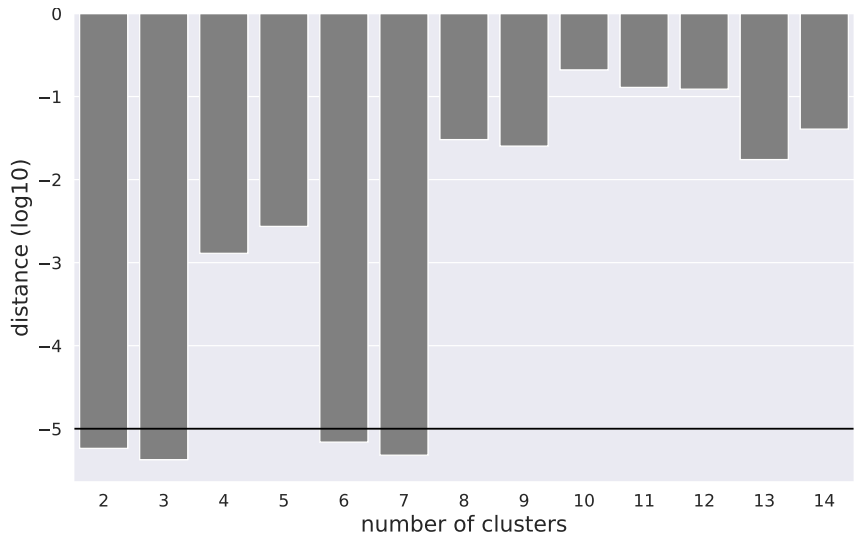


FIGURE 3.10: Convergence of CorEx applied to time courses obtained with 19-observable set through varied numbers of clusters. $\log_{10}$-transformed convergence scores (*Section 3.3.3*) of clusterings are represented with bars that if crossed a blue line are assumed as converged. The blue line is plotted on the level of $\log_{10}(1e-05)$. CorEx arrived at a stable solution for only four clusterings.

above the remaining clusters in almost all the clusterings. This suggests that prioritised observables most likely will belong to this cluster. Moreover, comparing the consistency of member allocation in the largest and strongest cluster can support the observed stabilisation of CorEx results and therefore, help to establish the final choice of the number of clusters.

In FIGURE 3.9A, agreement between pairs of the original clusterings can be observed between all clusterings in a rage starting from 7 to 12 clusters. This agreement in contents of the strongest cluster can be noticed for even lower numbers of clusters in the transformed clusterings (FIGURE 3.9B).

We can also examine the convergence metric (*Section 3.3.3*) over the change of cluster number. FIGURE 3.10 shows that the distance between mean values of last ten iterations visibly increases for more than 7 clusters. Despite this lack of convergence, agreement between clusterings was observed for clusterings with higher than 6 clusters.

We could observe that the number of clusters with at least one member stabilised to 8 for clusterings indicated to have 9 and more clusters. However, further examination of agreement in membership between clusterings showed that the top ARI scores were achieved for clusterings with lower numbers

of partitions, starting from 7 for the raw CorEx results, and already for the clustering of 3 partitions, when the consistency in membership was compared between the largest clusters. Although the safest value of clusters to choose for the 19-observable set is above 8, a firm and reasonably satisfying result can be obtained for 7 clusters.

The same analyses is also conducted for the 91-observable set. Figure 3.11 shows less stability in cluster size than observed in the 19-observable set. The maximal number of clusters with at least one associated observable is 29 but this number occurred only once for the clustering of 39 clusters. Similarly to the 19-observable set, there is a distinctive difference in the cluster strength and size between the first and the other clusters. Though it might not be that clear from the view on distribution of cluster sizes over varied number of clusters (Figure 3.11), the cluster strength distribution demonstrates this prevailing and wide discrepancy (Figures 3.12). Furthermore, for higher cluster indices, the ratio drops to 0 and lower down to negative values that indicates a lack of dependency between cluster members [253]. Therefore, despite sizes of clusters are larger than in the 19-observable set, far greater number of clusters has none or relatively low dependence between members, measured by the cluster strength. At the same time, there is a particular similarity between these observable sets as in both of them the normalised cluster strength of the largest cluster in nearly all clusterings is equal 0.4 (compare Figures 3.13 & 3.8C). These observations can be explained by the fact that the 91 observable set is composed of far more fined grained output than the 19-observable set, drawn from the same system but with potentially higher level of noise. In this particular dataset, the noise would denote these molecular species that appear in very low quantities or very rarely over the simulation. In majority of clusterings of the 91-observable set, the largest cluster has 15 observables with the cluster strength of 8, whereas an equivalent cluster in the 19-observables has 8 observables with the cluster strength of 4. Recalling that the maximal value of the cluster strength is $m-1$, where m is the cluster size, both cases represent quite robust signals [253]. CorEx filtered out noisy and weakly dependent observables from the time courses of the 91-observable dataset and found proportionally the same amount of information in time courses of both datasets.

Figures 3.14 shows agreement of cluster membership allocation mea-

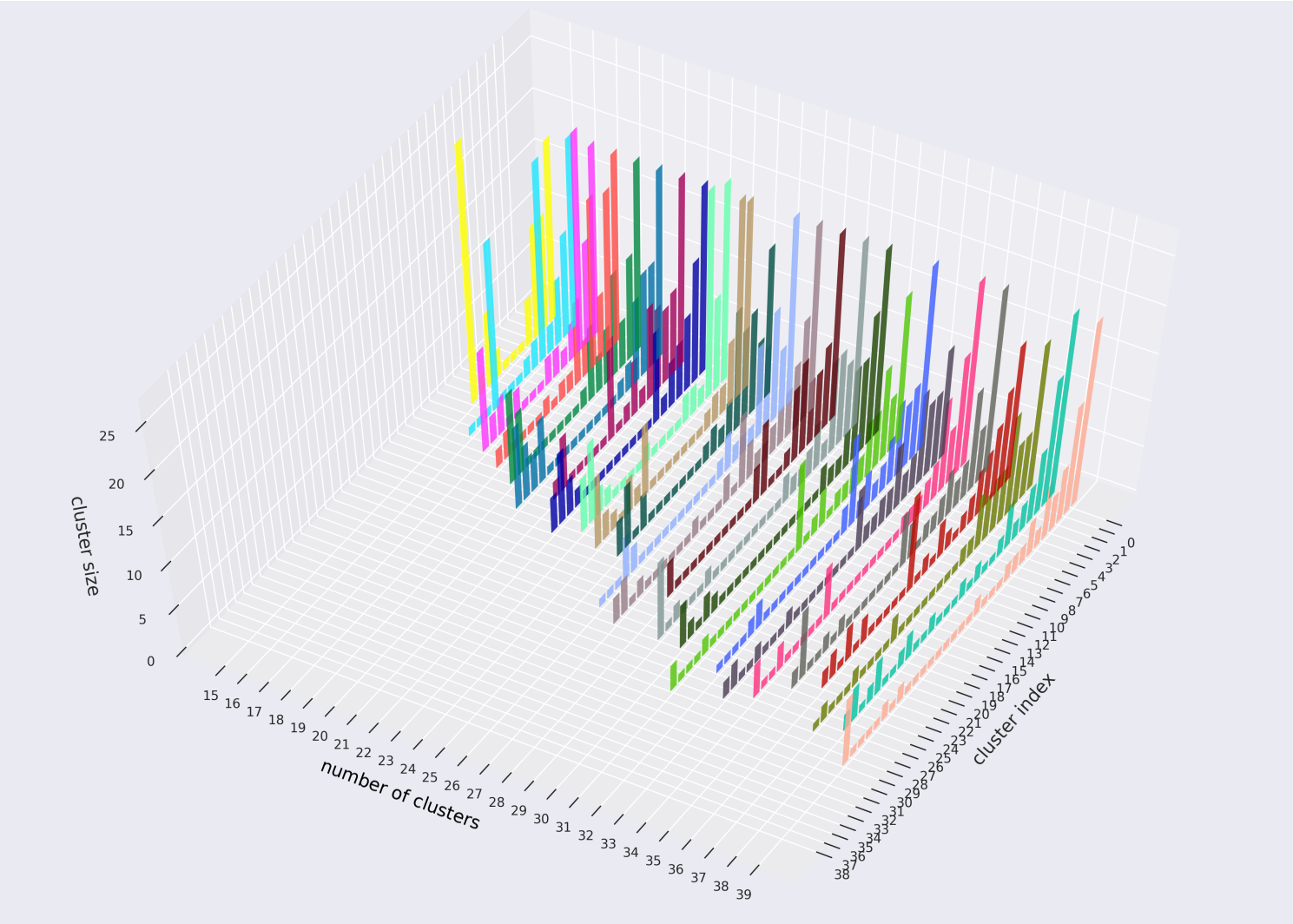


FIGURE 3.11: Change in cluster sizes plotted against increasing number of clusters for clusterings obtained with time courses of the 91-observable set. The maximal number of clusters that have at least one associated observable is 29 seen in the clustering of 39 clusters. Variation in non-empty clusters is much larger then in the 19-observable set.

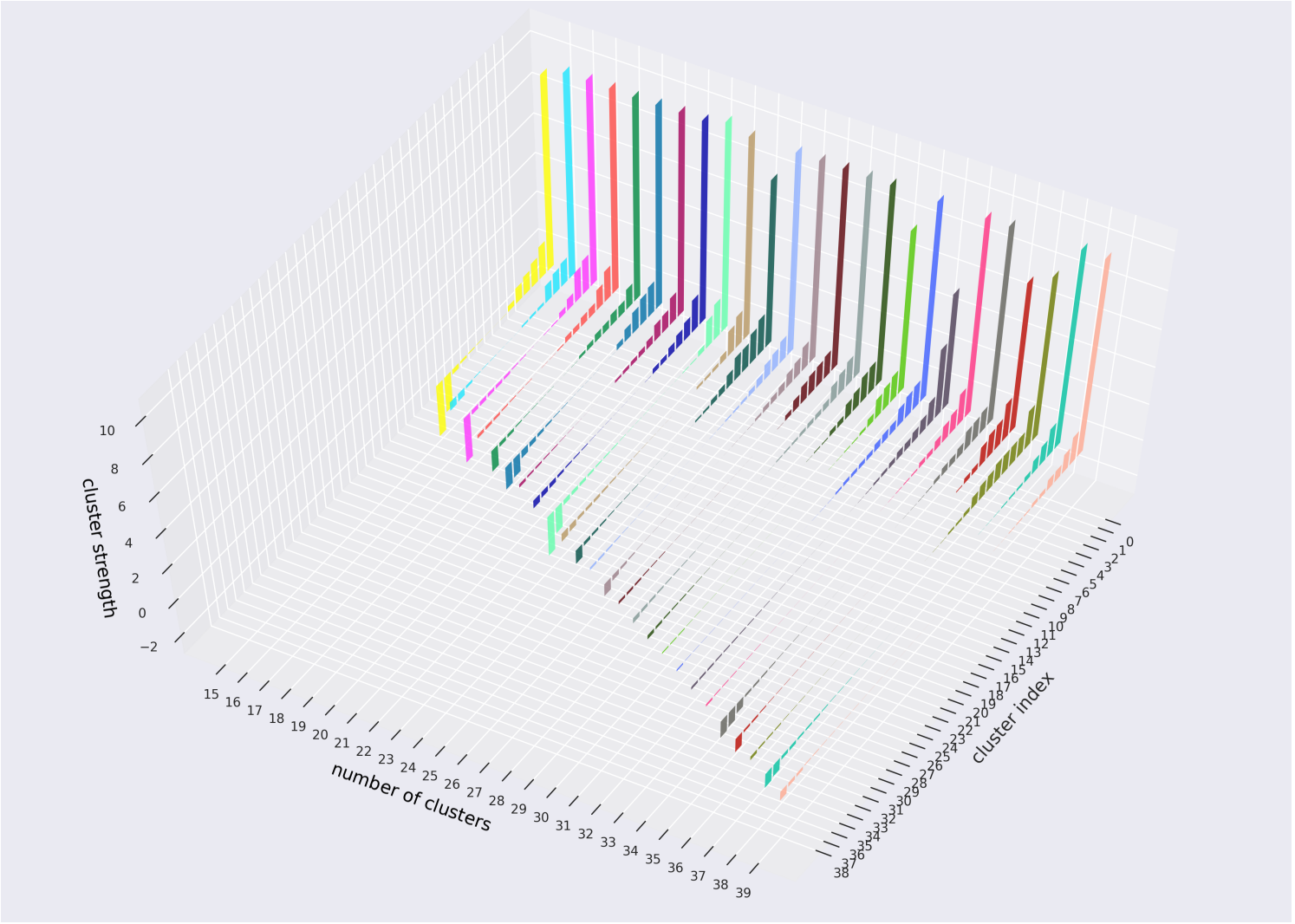


FIGURE 3.12: Change in cluster strengths plotted against increasing number of clusters for clusterings obtained with time courses of the 91-observable set. Distinctive discrepancy between the first strongest cluster and the remaining clusters can be observed. The value of strength for this cluster is repeated in almost all clusterings.

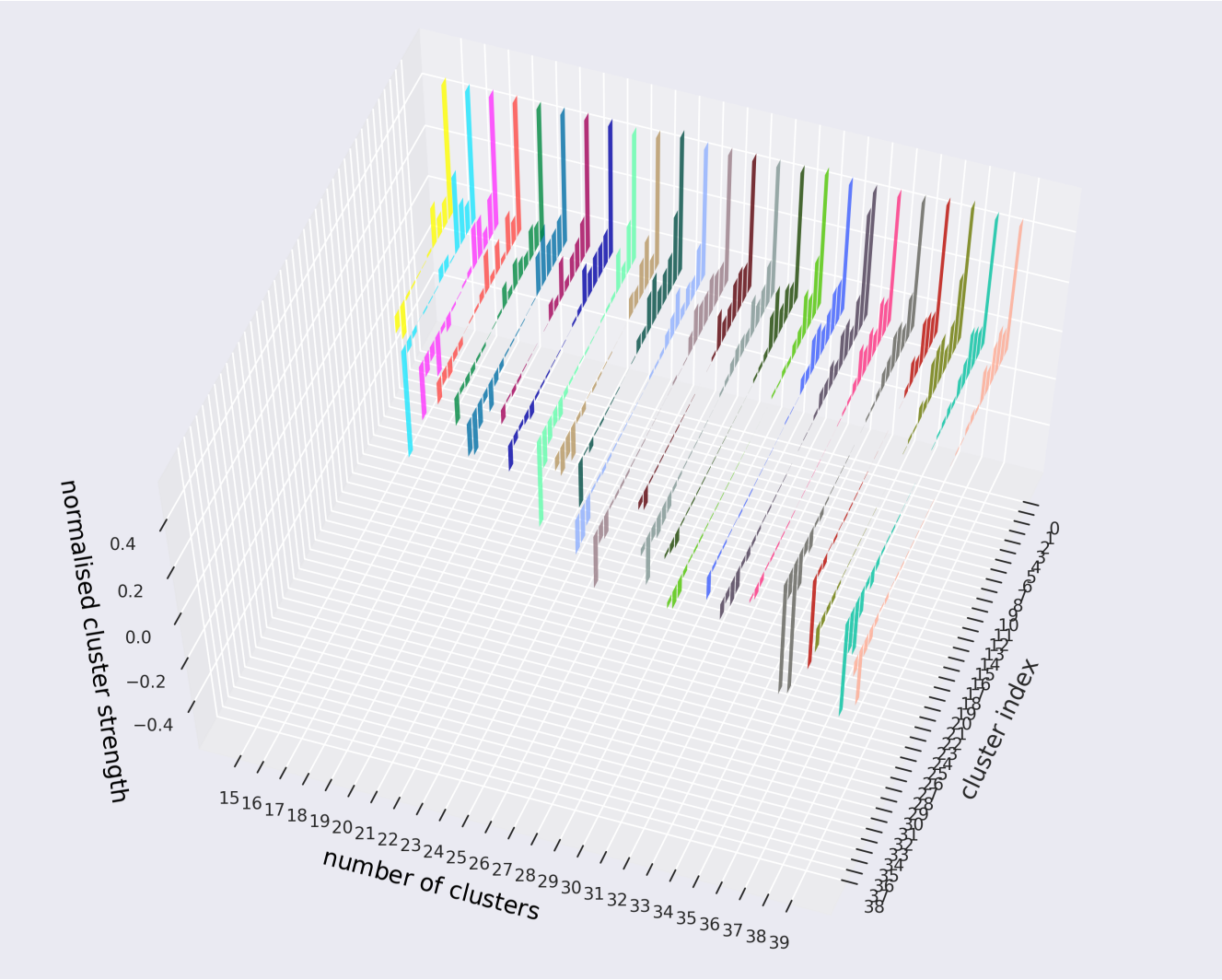


FIGURE 3.13: Normalised cluster strength plotted against increasing numbers of clusters for clusterings obtained with time courses of the 91-observable set. The normalised cluster strength for the strongest clusters has the same value of 0.4 as in the 19-observable set for all clusterings.

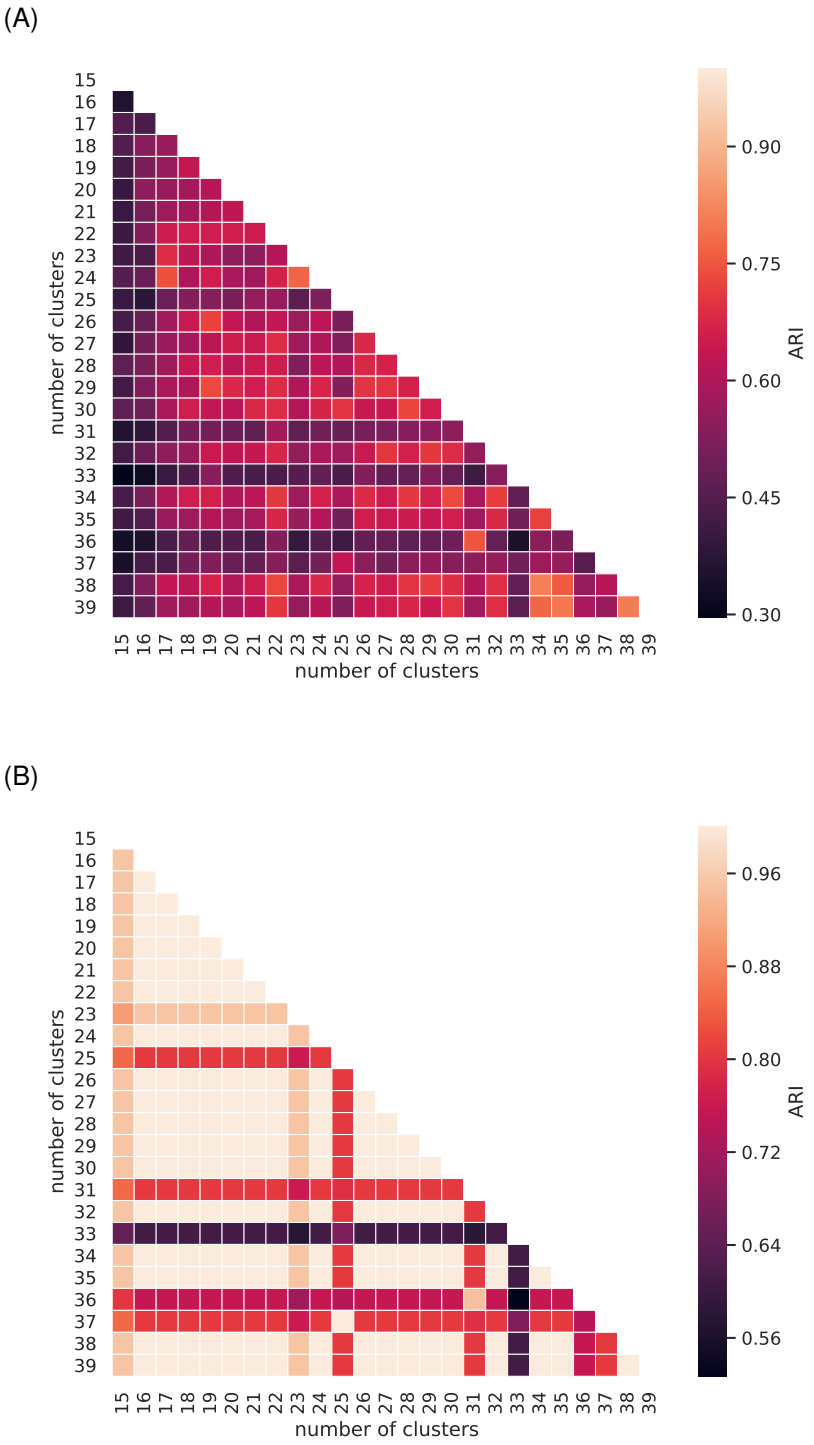


FIGURE 3.14: Measurement of agreement in observable allocation between clusterings with increasing numbers of clusters computed for time courses of the 91-observable set. The agreement between pairs of clusterings was calculated with ARI, for original clusterings (A), and transformed clusterings (B), composed of two clusters, where the first one has members that belong to the largest and the strongest cluster, and the second one contains members from other clusters.

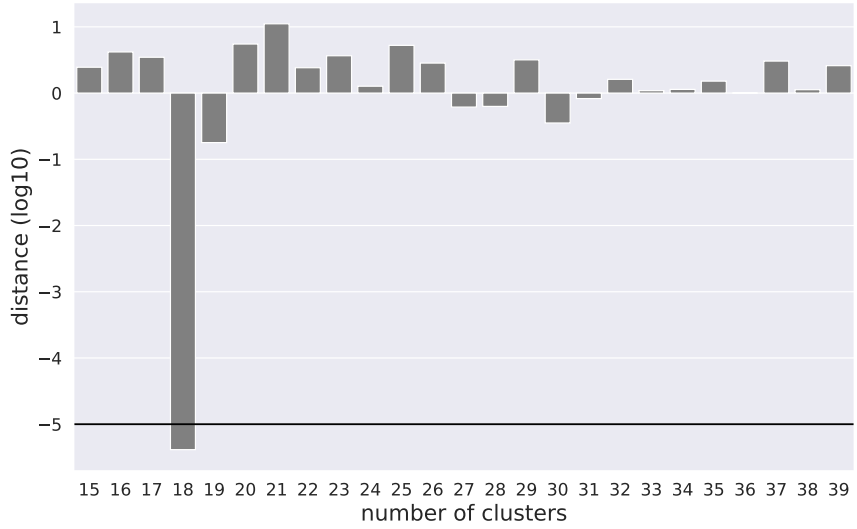


FIGURE 3.15: Convergence of the 91-observable set over varied numbers of clusters. $\log_{10}$-transformed convergence scores (*Section 3.3.3*) of clusterings are represented with bars that if crossed a blue line are assumed as converged. The blue line is plotted on the level of $\log_{10}(1e-05)$. CorEx arrived at a stable solution for only one clustering with 18 clusters. Compared to the 19-observable set, clustering of the 91-observable set is highly unlikely to converge. This can be explained by higher level of more fragmented representation of signal in the 91-observable set.

sured with ARI between pairs of clusterings with increasing number of clusters. FIGURE 3.14A shows results for all paired clusterings and FIGURE 3.14B shows results for transformed clustering determining consistency in the first cluster membership across all clusterings. In the former figure, there is a very low agreement between clusterings compared to the 19-observable set that can be explained with the presence of molecular species with very low abundances. Complete identity between clusterings can be found only when the first clusters are compared, that is preserved fairly consistently over different numbers of clusters. Suggested by these observations, the most reasonable choice of the number of clusters for this observable set is above 15.

Convergence of CorEx was also recorded for the 91-observable set demonstrated in FIGURE 3.15. Results show much larger distances between learned values of $\sum TC(X;Y_j)$ than observed in the 19-observable set. Similarly to the 19-observable set, the lack of convergence did not hinder the ability of CorEx to indicate the most tightly interlinked subgroup of observables. This can be argumented by an example of clustering with 21-clusters, represented

by the highest bar in FIGURE 3.15. Despite this clustering is most distant from convergence, FIGURE 3.14B demonstrates that CorEx identified members of the first cluster with a complete agreement with other clusterings.

To conclude, multiple CorEx evaluations of time courses with two different observables sets, performed with gradually higher number of clusters, indicated that the most reasonable range of values for this input parameter for the 19-observable set is above 6, and for the 91-observable set, above 15. These values are inferred mainly on the membership agreement between clusterings. As exemplified by the 91-observable set that contains trajectories characterised with weak signals, the increase of cluster numbers did not bring stability in the number of clusters filled with at least one observable. Therefore, the stability of the largest cluster between clusterings seemed to be more indicative.

Analysis of obtained clusterings hinted on general characteristics of CorEx results with respect to these particular datasets. As seen, the clustering of time courses of the 91-observable data set is less stable over multiple runs than the clustering of time courses of the 19-observable data set. This can be explained by the amount of noise in the former data set. The 91-observable data set is in fact a more fragmented representation of the model than the 19-observable data set. The same prevailing value of normalised cluster strengths, allocated for the strongest clusters, indicates that CorEx identified equal proportion of signal in time courses of respective observable sets.

Procedure of learning dependencies within in a dataset by CorEx is not deterministic as in a few clusterings a fixed value of total correlation between iterations was not successfully found, even for the 19-observable data set. This justifies application of CorEx to all time courses obtained with varied parameter samples to prioritise observable sets, that is the main objective of this pipeline.

Despite being only guaranteed to converge to local optimum, CorEx is able to identify a single and strongly dependent cluster of observables, that is consistent over majority of multiple runs with increasing numbers of clusters. This strongest cluster was successfully identified regardless learning convergence. The other clusters have distinctively weaker dependence what suggest that in further pipeline steps, the focus should be placed on the strongest cluster that potentially harbour the most information within both observable data sets. To confirm validity of this largest cluster, observables that are its members should be examined with respect to their relations encoded in the model. This

is the subject of the next section, where observables of the strongest clusters found in both observable data sets are studied in details.

3.4.2 Comparison of clusterings between selected and sampled observables

Having found suitable settings for both observable sets, we can examine and compare allocation of observables within clusterings of these sets. CorEx executions were automatically repeated 3 times and the highest value of $\sum TC(X;Y_j)$ was subjected to analysis.

Automated visualisation of clustering are provided with tree graphs as seen in Figure 3.16 and 3.17. A cluster is a circle node with outgoing edges to observable names that are cluster members. The number placed in the middle of nodes is the cluster index. The size of nodes is proportional to the cluster strength (TC_j). Thickness of outgoing edges between nodes and observable names is proportional to the observable strength (MIS_{ji}) defining strength of dependence between a cluster and observable.

Figure 3.16 demonstrates outcomes of clustering performed on time courses of the 19-observable set. The number of clusters was set to 7. The largest 9-element cluster with the 0th index is also the strongest one, scoring $TC_j = 4.428$. The remaining 10 observables are scattered between other 6 clusters. Table 3.2A enlists names of these 9 members together with their observable strength (MIS_{ji}) in descending order. As seen, the values of MIS_{ji} are not distinctively different. The 9 observables can be represented with 6 observables denoting the same molecular species but in a more generalised form (Table 3.2B). This reduction is achieved by removing redundant context (binding or internal states of sites) in expressions representing observables of the same agent. Molecular species in 3rd, 5th, 6th and 7th rows of Table 3.2A can be replaced with an observable "D34", denoting DARPP-32 phoshophorylated at Threonine 34 (Thr34) with unspecified binding state and internal states of two other sites. "D34" belongs to the list of 15 observables defined in *Table 2.1* (*Chapter 2*), that represented species of DARPP-32 as overlapping sets. However, this list of 19 observables with the explicit and non-overlapping representation of DARPP-32 species demonstrates that species with phosphorylation of the Thr75 site cannot be fussed in its general form (i.e. "D75") as one of species that would implicitly be covered by this representation, "Thr75:Ser137", is

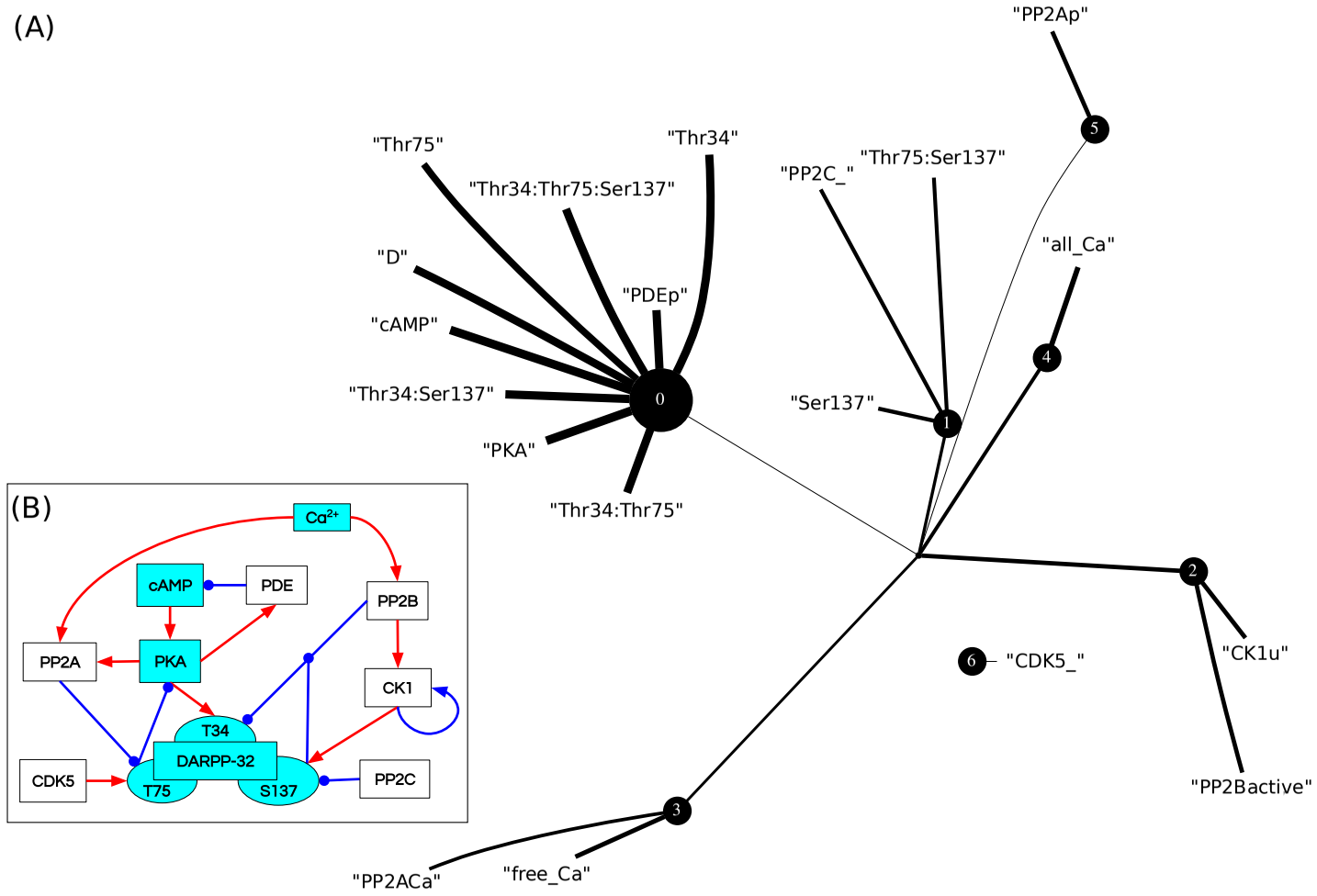


FIGURE 3.16: CorEx clustering result visualised with a tree graph for the 19-observable set (A). Nodes are clusters with edges outgoing to observable names. Cluster indices are marked with integers on each node. Node size is proportional to cluster strength. The 0th-indexed cluster is the strongest and the largest one. Observables of this cluster are marked in cyan on the reaction diagram of the DARPP-32 network model (B) indicating a cAMP-activated subnetwork. Refer to TABLE 2.1 and *Section 3.3.2* for definitions of observables.

(A)

	Observable name	Observable expression in Kappa language	MIS_{ji}
1.	PKA	PKA()	0.603
2.	cAMP	cAMP(c~on?)	0.592
3.	Thr34:Ser137	D(thr34~p, ser137~p, thr75~u)	0.575
4.	PDEP	PDE(pSite~p?)	0.572
5.	Thr34:Thr75:Ser137	D(thr34~p, ser137~p, thr75~p)	0.568
6.	Thr34	D(thr34~p, ser137~u, thr75~u)	0.567
7.	Thr34:Thr75	D(thr34~p, ser137~u, thr75~p)	0.566
8.	D	D(thr34~u, ser137~u, thr75~u)	0.549
9.	Thr75	D(thr75~p, thr34~u, ser137~u)	0.478

(B)

	Observable name	Observable expression in Kappa language	Row(s) in TAB. 3.2A
1.	PKA	PKA()	1
2.	cAMP	cAMP(c~on?)	2
3.	PDEP	PDE(pSite~p?)	4
4.	D	D(thr34~u, ser137~u, thr75~u)	8
5.	Thr75	D(thr75~p, thr34~u, ser137~u)	9
6.	D34	D(thr34~p)	3,5,6,7

TABLE 3.2: List of observables assigned to the largest and strongest cluster ($TC_j = 4.428$) found with time series of the 19-observable set (A). This is the most visible cluster on FIGURE 3.16. Molecular species in 3rd, 5th, 6th and 7th rows can be represented with a single observable "D34", denoting DARPP-32 phoshophorylated at Thr34 with unspecified binding and internal states of other two site. Therefore, the list of 9 observables can be reduced to a 6-observable list (B). Species phosphorylated at the Threonine 75 (Thr75) site cannot be similarly generalised as "D34", because "D75" implicitly includes "Thr75:Ser137" that is assigned to a different cluster.

not present in the largest cluster. This missing species can be located in the cluster with the index 1 (Figure 3.16), together with DARPP-32 phosphorylated only at the Serine 137 (Ser137) site (observable "Ser137"), and the bound form of protein phosphatase 2C (PP2C) dephosphorylating Ser137 (observable "PP2C_").

All species of DARPP-32 phosphorylated at Thr34 appeared in the 0-indexed cluster indicating on importance of events involving Thr34. Among them are observables representing a negative feedback loop (cAMP–protein kinase A (PKA)–PDE, Figure 3.16B) regulating the level of cAMP. This feedback loop is directly involved in the phosphorylation of Thr34 site as are other cluster members. This observable allocation indicates that CorEx separated major effectors of the cAMP signal, from the ones. Presence of unphosphorylated DARPP-32 in this cluster can be explained by the fact that the unphosphorylated DARPP-32 is mainly used for the phosphorylation of Thr34. This can be supported by the observation of the opposite dynamics of the unphosphorylated DARPP-32 and the DARPP-32 phosphorylated at Thr34 that react to the cAMP signal (Figure 2.10).

Similar analysis can be performed for clustering of time courses of the 91-observable set (Figure 3.17). The number of clusters was set to 18. The largest 20-element cluster is also the strongest one. The content of the strongest cluster with the index 0 is enlisted in Table 3.3 and presented together with MIS_{ji} values defined per observable in descending order. There is slightly larger discrepancy between these values than in the cluster of 19-observables set as the top scored observable (1st row) has a doubled value of the observable that got the lowest score (the last row). The largest cluster strength is much higher then its counterpart in the clustering of the 19-observable set, that is $TC_j = 10.478$. Similarly to the previous observable set, the list of 20 observables in Table 3.3, can be reduced to 12 observables, as shown in Table 3.4.

As in the 19-observable set, majority of molecular species in the largest cluster of the 91-observable set are directly related to the cAMP signal, such as PKA, PDE, cAMP, R2C2 and DARPP-32 phosphorylated at Threonine 34 (D34). In case of the 91-observable set, however, we have a view on explicitly defined observables that gives us precise information what particular molecular species, including complexes, are important and dependent on each other. For instance, cAMP is found as both unbound and bound to R2C2. If four

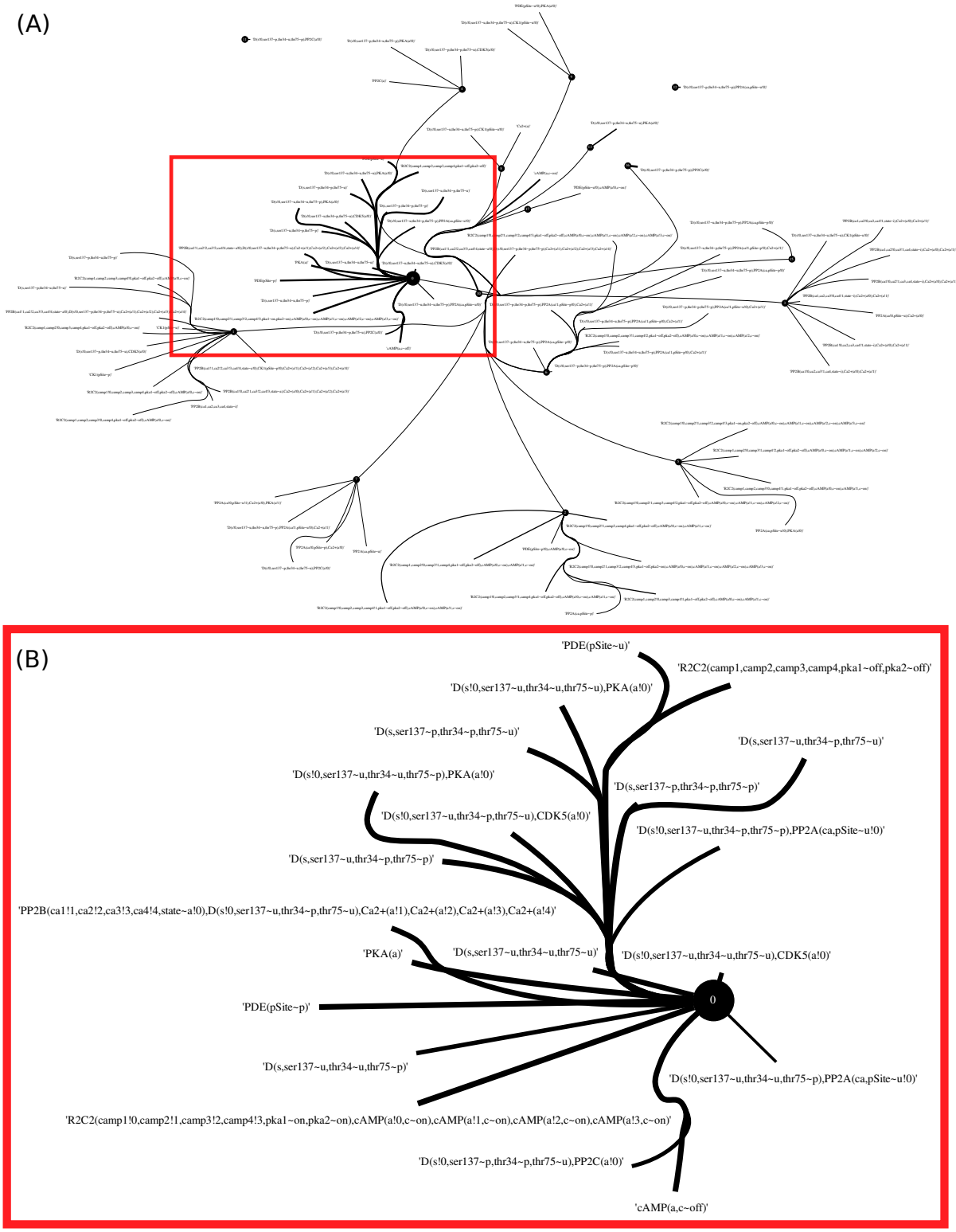


FIGURE 3.17: Clustering of the 91-observable set visualised with a tree graph (A). Graph nodes are clusters with edges outgoing to observable names. Node size is proportional to cluster strength (TC_j). CorEx assigned observables to 17 clusters, with one particularly stronger than the others (B), composed of 20 observables with similarly strong observable-to-cluster dependence (MIS_{ji}) proportional to thickness of edge strokes.

	Observable expression in Kappa language	MIS
1.	D(s!0, ser137~u, thr34~u, thr75~u), PKA(a!0)	0.622
2.	R2C2(camp1!0, camp2!1, camp3!2, camp4!3, pka1~on, pka2~on), cAMP(a!0, c~on), cAMP(a!1, c~on),cAMP(a!2, c~on),cAMP(a!3, c~on)	0.616
3.	R2C2(camp1, camp2, camp3, camp4,pka1~off, pka2~off)	0.616
4.	PKA(a)	0.615
5.	D(s!0, ser137~u, thr34~u, thr75~p), PKA(a!0)	0.601
6.	PDE(pSite~u)	0.599
7.	PDE(pSite~p)	0.599
8.	D(s, ser137~u, thr34~p, thr75~u)	0.598
9.	cAMP(a, c~off)	0.598
10.	D(s, ser137~u, thr34~p, thr75~p)	0.598
11.	D(s, ser137~p, thr34~p, thr75~u)	0.597
12.	D(s, ser137~p, thr34~p, thr75~p)	0.587
13.	D(s!0, ser137~u, thr34~p, thr75~u), CDK5(a!0)	0.585
14.	D(s, ser137~u, thr34~u, thr75~u)	0.574
15.	PP2B(ca1!1, ca2!2, ca3!3, ca4!4, state~a!0), D(s!0, ser137~u, thr34~p, thr75~u), Ca2+(a!1), Ca2+(a!2), Ca2+(a!3), Ca2+(a!4)	0.571
16.	D(s!0, ser137~u, thr34~u, thr75~u), CDK5(a!0)	0.503
17.	D(s!0, ser137~u, thr34~p, thr75~p), PP2A(ca, pSite~u!0)	0.478
18.	D(s, ser137~u, thr34~u, thr75~p)	0.472
19.	D(s!0, ser137~p, thr34~p, thr75~u), PP2C(a!0)	0.428
20.	D(s!0, ser137~u, thr34~u, thr75~p), PP2A(ca, pSite~u!0)	0.305

TABLE 3.3: List of observables assigned to the largest and strongest cluster ($TC_j = 10.478$) found with time series of the 19-observable set. This is the most visible cluster on FIGURE 3.17.

	Observable expression in Kappa language	Row(s) in Tab. 3.3
1.	D(s!0, ser137~u, thr34~u),PKA(a!0)	1,5
2.	R2C2(camp1!0,camp2!1,camp3!2,camp4!3,pka1~on,pka2~on), cAMP(a!0,c~on),cAMP(a!1,c~on),cAMP(a!2,c~on),cAMP(a!3,c~on)	2
3.	R2C2(camp1,camp2,camp3,camp4,pka1~off,pka2~off)	3
4.	PKA(a)	4
5.	PDE(pSite)	6,7
6.	D(s, thr34~p)	8,10,11,12
7.	cAMP(a, c~off)	9
8.	D(s!0, ser137~u,thr75~u),CDK5(a!0)	13,16
9.	D(s, ser137~u,thr34~u)	14,18
10.	PP2B(ca1!1,ca2!2,ca3!3,ca4!4,state~a!0), D(s!0,ser137~u,thr34~p,thr75~u), Ca2+(a!1),Ca2+(a!2),Ca2+(a!3),Ca2+(a!4)	15
11.	D(s!0,ser137~u,thr75~p),PP2A(ca,pSite~u!0)	17,20
12.	D(s!0,ser137~p,thr34~p,thr75~u),PP2C(a!0)	19

Table 3.4: Reduced list of 12 observables assigned to the strongest cluster of the 91-observable set. The reduction of the observables list is performed by representing multiple observables as one generalised expression. Row numbers of combined observables from the complete list of 20 observables in Table 3.3 are enlisted in the third column.

molecules of cAMP are bound to R2C2, it can cause dissociation of two PKA kinases. Furthermore, complexes of DARPP-32 with PP2C, protein phosphatase 2 (PP2A) and protein phosphatase 3/calcineurin (PP2B), cyclin dependent kinase 5 (CDK5) and PKA are important only when DARPP-32 is in specific configuration of phosphorylation sites. For instance, in the complex of DARPP-32 and CDK5 (the 8th row in Table 3.4), DARPP-32 is unphosphorylated on Ser137 and Thr75 sites. The state of Thr34 is irrelevant as in the table with unreduced observable list it appears as unphosphorylated and phosphorylated (rows 13th and 16th Table 3.4). The reason why DARPP-32 is unphosphorylated at the Thr75 site in the complex of DARPP-32 and CDK5 is that CDK5 phosphorylates Thr75 site that has to be dephosphorylated to bind CDK5 as there is not product rebinding. Similar explanation of explicit state indication of a particular phosphorylation sites applies in three other complexes that involves DARPP-32 (1st, 8th, 11th rows in Table 3.4). In the largest cluster, there is no observables of bound and fully phoshophorylated form of DARPP-32. In particular the Ser137 site is in nearly all such cases unphosphorylated, except for a complex of DARPP-32 and the phosphatase PP2C of Ser137 (12th row of Table 3.4). Reasons why these specific forms of molecular species and site configurations appeared in the dominating cluster but not the others with very similar composition might be related to their much higher abundances or that they are largely independent from each other. The larger abundances of certain species might be caused by a lower number of reaction steps necessary to create these species, leaving more complex species in marginal abundances. For instance, in the 11th row of Table 3.4 showing a complex of DARPP-32 and PP2A, PP2A is not bound to Ca^{2+} nor phosphorylated that would require execution of two additional reaction rules. On the other hand, increase in molecular species of PP2A bound to Ca^{2+} require the Ca^{2+} spiking, the signal by which affected species seem to not be represented in the largest cluster. The only observable that contains Ca^{2+} ions is the observable in the 10th row of Table 3.4 that represents a complex of DARPP-32, phosphorylated at Thr34, and PP2B, a phosphatase dephosphorylating Thr34. According to the rule specification, only an active form of PP2B, bound to four Ca^{2+} ions, can dephosphorylate Thr34. Despite the presence of Ca^{2+} ions, abundances of this particular species are very much dependent on the cAMP-triggered signal and the phosphorylation of Thr34.

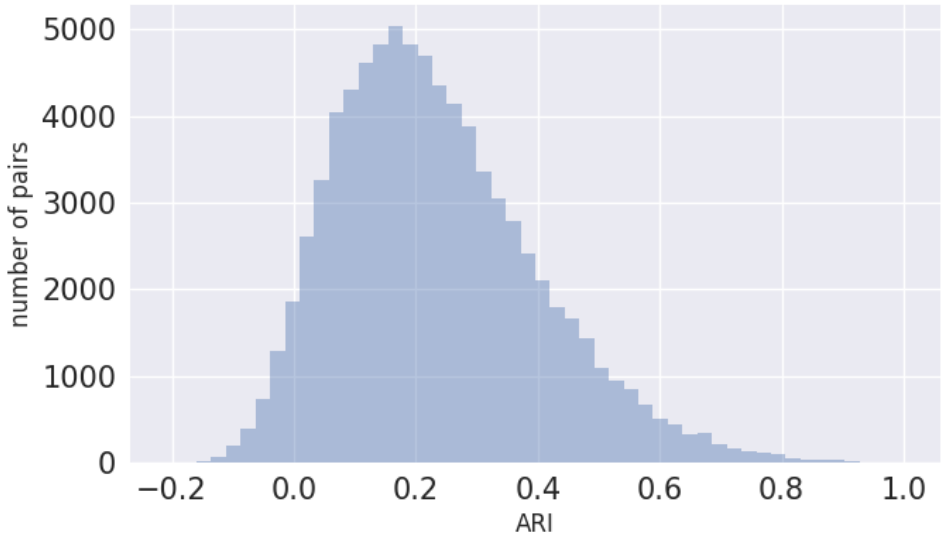
Closer examination of observables assigned to the dominating cluster in both observable data sets indicates that the cluster is constructed from members directly affected by the cAMP signal. Differently to the hand-selected and generalised 19-observable data set, the automatically generated 91-observable set is a list of exact compositions and configurations of molecular species that appeared during the simulation. Hence, the largest cluster derived from clustering of time courses of the 91-observable set contains exact information on species configuration that are strongly dependent on each other. We learned that only particular configurations of species are found in the cluster and therefore, are strongly dependent on the cAMP-signalling events. This observation suggests that not all closely matching species are present in equal abundances during the simulation, or their dependence on other signal in the model is much stronger than to the cAMP signal. Nevertheless, the original lists of observables in both sets were reduced to more generic representation of observables by removing redundant context in their expressions. This supports the assertion that CorEx detected equally abundant and closely matching observables that brings to the light another facet of CorEx application, that is dimensionality reduction.

3.4.3 Observable scores

Acquiring a view on clustering results produced by CorEx application on single time courses, we can delve into results obtained with the pipeline (Figure 3.4), where CorEx is applied to 400 averaged time courses collected from simulations of the model with varied parameter sets. This section presents calculation outcomes of two observable scores to clusterings obtained with CorEx. The first observable score is defined with respect to the clustering type and therefore, contains a frequency term of clustering type. The second observable score is calculated by using CorEx output measures form all clusterings of parameter samples without differentiating them into types. As these observable scores are sums of more than one CorEx output measures, results of each one is analysed in separation. To simplify analyses, the pipeline is applied to time courses of the 19-observable set. The parameters of the model are varied 10 folds from their fundamental values.

Averaged time courses obtained with model run performed with different parameter set are clustered with CorEx with the number of clusters set to

(A)



(B)

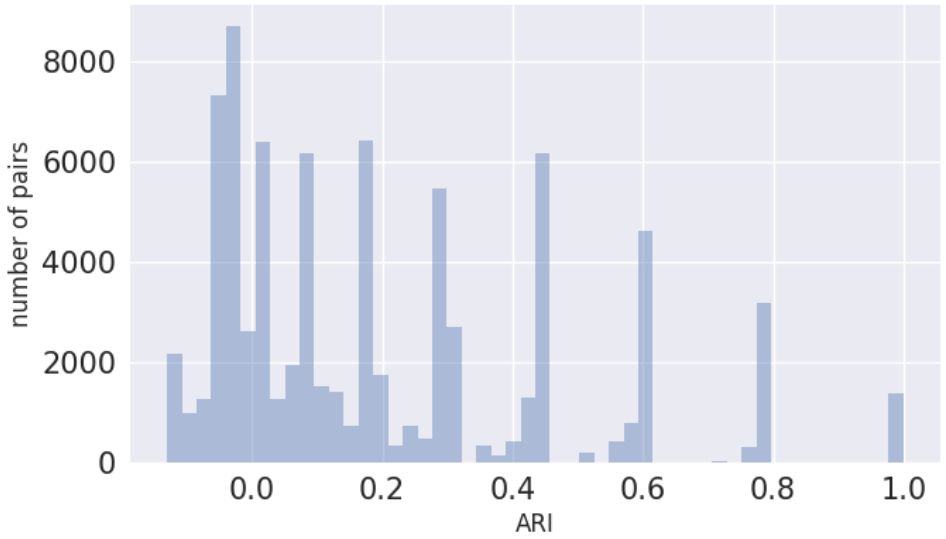


Figure 3.18: Distribution of ARI scores for pairs of clusterings obtained by applying CorEx to time courses generated with all parameter sample sets. When clustering pairs are compared in their original number of clusters as partitioned by CorEx, there is weak agreement between $79\,800$ clustering pairs except for one pair that scored 1 (A). When only content of the first cluster is compared between clustering, the number of identical clustering pairs reaches 1377 comprising 1.73% of all clustering pairs (B). This low fraction of repeated clusterings reveals absence of any types of clusterings in time courses of the 19-observable set generated with 10 fold parameter variations.

10. To determine whether any repeated clusterings types can be identified, Adjusted Rand Index (ARI) is calculated for each pair of clusterings. Figure 3.18A shows that great majority of scores calculated between 79800 pairs have very low similarity, located between 0.0 and 0.4, with the mean value slightly below 0.2. What is more, there are only two clusterings which scored 1 meaning that only one pair of clusterings is identical. When comparison of only first clusters between clusterings is performed, as in *Section 3.4.1*, the number of identical clustering pairs that scored 1 is 1377, that comprises 1.73% of all pairs (Figure 3.18B). This low fraction of identical clusterings seems to be found rather by chance and therefore, does not convey existence of types among clusterings in the dataset that would divide clusterings into groups of identical clusterings. This lack of distinctive agreement between groups of clusterings might be caused by a relatively large range of 10 fold parameter variation that significantly perturbed the modelled system. Therefore, following analysis concentrates on constituents of Equation 3.16 that takes into account all clustered samples without the frequency term calculated per clustering type. The frequency term is excluded from the equation for the observable score as no clustering types were identified.

Figure 3.19 shows distribution of observable counts per cluster. Clusters included in the figure, are derived from clusterings of time courses generated with all parameter sets. With the maximal number of clusters set to 10 and clustering of 400 time courses, a total number of clusters is 4000. Figure 3.19 shows that nearly 2400 clusters, that is more than half of all clusters, have less than two cluster members that classifies them to degenerate clusters.

Figures 3.20 show distributions of cluster strengths (TC) for all clusterings. Figure 3.20A shows raw TC values that take values between 0 to above 6. Cluster strength of 2200 clusters is equal 0 denoting degenerated clusters. Figure 3.20B shows normalised values of TC, TC', enclosed within 0 and 0.6. The normalisation is performed by dividing TC values by member counts if their values are > 0 and cluster member count is > 1. Otherwise, TC' values are set to 0. After the procedure of normalisation, the number of clusters with strength equal to 0 increased by 200 to the total of 2400 degenerate clusters. Grouping together all degenerate cluster strengths under the 0 score reveals a bimodal distribution of remaining cluster strengths with the first mode located between 0.1 and 0.2 values of TC', and the second above the value of 0.5.

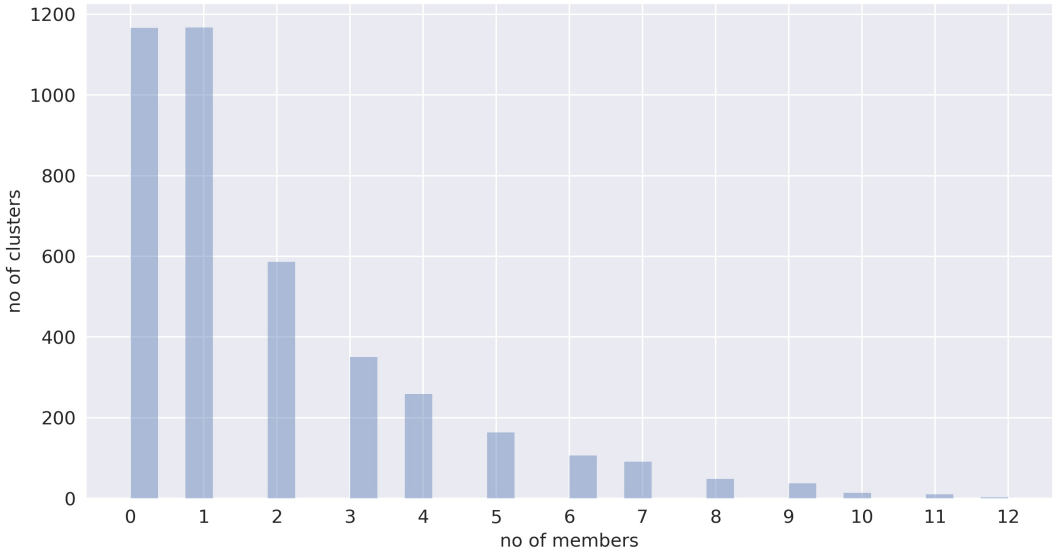
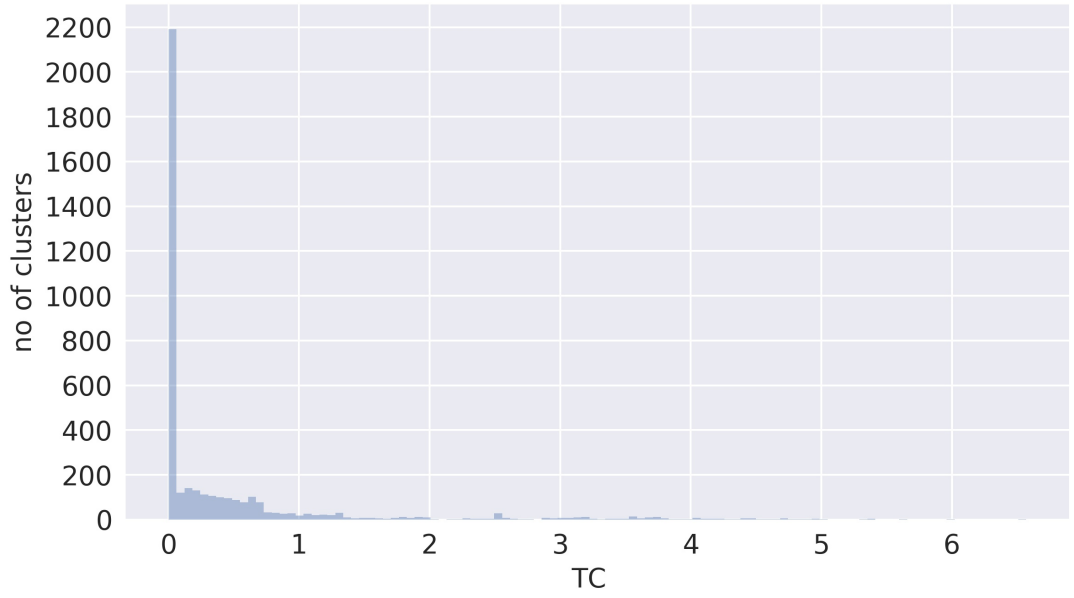


FIGURE 3.19: Distribution of member counts per cluster. Clusters that are included are derived from clusterings of time courses generated with all parameter sets. Singleton clusters, as non-member clusters, are classified as degenerate clusters in observable score definitions what amount to almost 2400 clusters with 0 or 1 member.

Cluster strengths are parcelled into observables to separately examine their allocation for each observable, for raw *TC* values (FIGURE 3.21) and normalised ones (FIGURE 3.22). For each observable, *TC* and *TC'* can take values from the whole range. For *TC* between 0 and 7, and for *TC'* between 0 and 0.6. In both cases, distribution of values is not uniform but tend to have unimodal or bimodal shape. In distributions of raw *TC* values, both types of distributions have the mode between 0 and 1. In the bimodal ones the second mode is located between 3 and 4. The observables that belong to the bimodal group are "Thr34:Thr75" (A), "cAMP" (B), "D" (E), "Thr34" (F), "PKA" (H), "Thr75" (J) "PDEp" (N). These 7 were among 9 located in the dominating cluster found by clustering of a single set of time courses generated with the baseline parameter set (*Section 3.4.2*, FIGURE 3.16). Multiple clusterings of time courses with parameter perturbations revealed that this earlier result identifying the 7 observables as strongly dependent is rather a tendency. Normalisation of values reduced broadness of variation and exposed more observables with bi- and tri-modalities in *TC'* distributions (FIGURE 3.22). These three

(A)



(B)

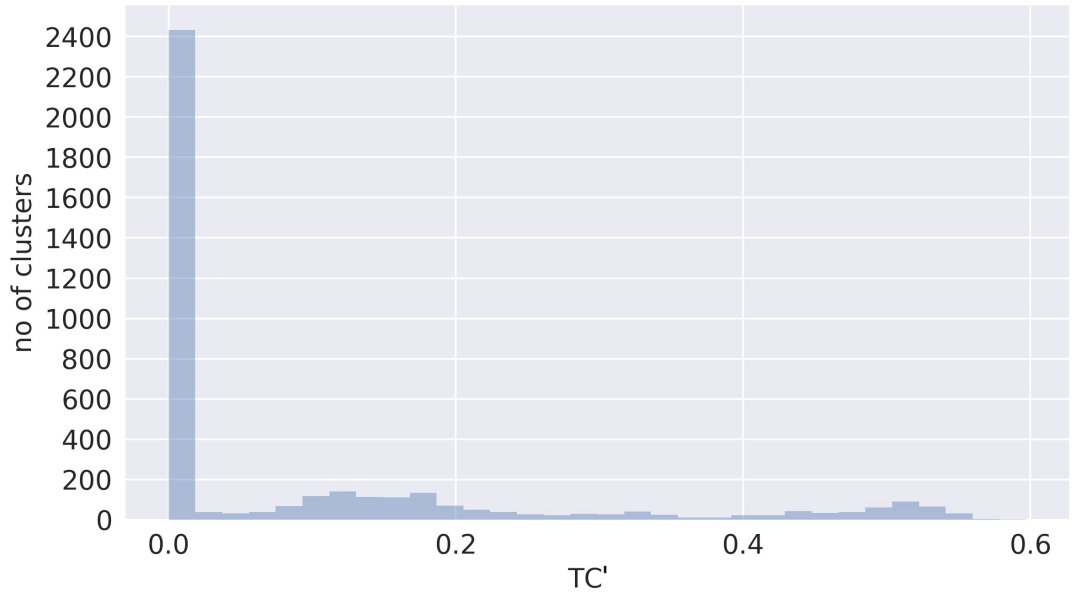


FIGURE 3.20: Distributions of the cluster strength (TC) derived from all clusterings obtained with time courses generated with varied parameter sets: (A) raw TC values, (B) normalised TC' values. Normalisation of TC values is performed by dividing TC values by member counts if their TC values are > 0 and member counts is > 1. Otherwise, TC' values are set to 0. Joining all degenerate cluster TC values under 0 score reveals a bimodal distribution of remaining cluster strengths with the first peak located between 0.1 and 0.2 values of TC', and the second above the value of 0.5.

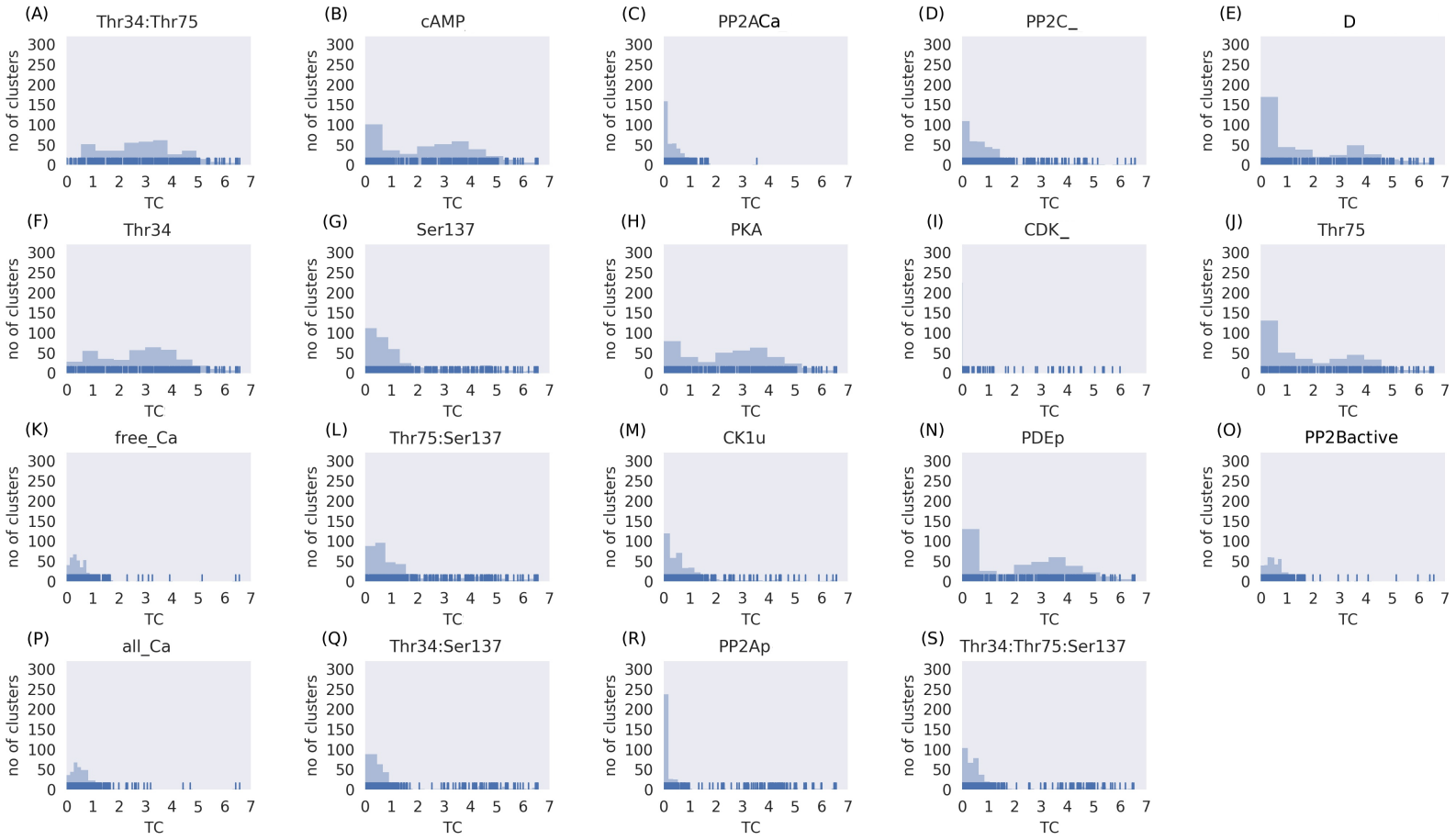


FIGURE 3.21: Distribution of raw cluster strengths (TC) per observable. Individual data points are displayed as rug plots superimposed on histograms. There are 7 observables with bimodal distributions that are frequently associated to clusters with TC values located between 3 and 4, that is the highest modality in distributions. These are “Thr34:Thr75” (A), “cAMP” (B), “D” (E), “Thr34” (F), “PKA” (H), “Thr75” (J) “PDEp” (N). These observables were among 9 located in the dominating cluster earlier discussed in *Section 3.4.2* and presented as a tree plot FIGURE 3.16. Multiple clusterings of time courses with parameter perturbations shows that strong dependency between 7 observables is rather a tendency.

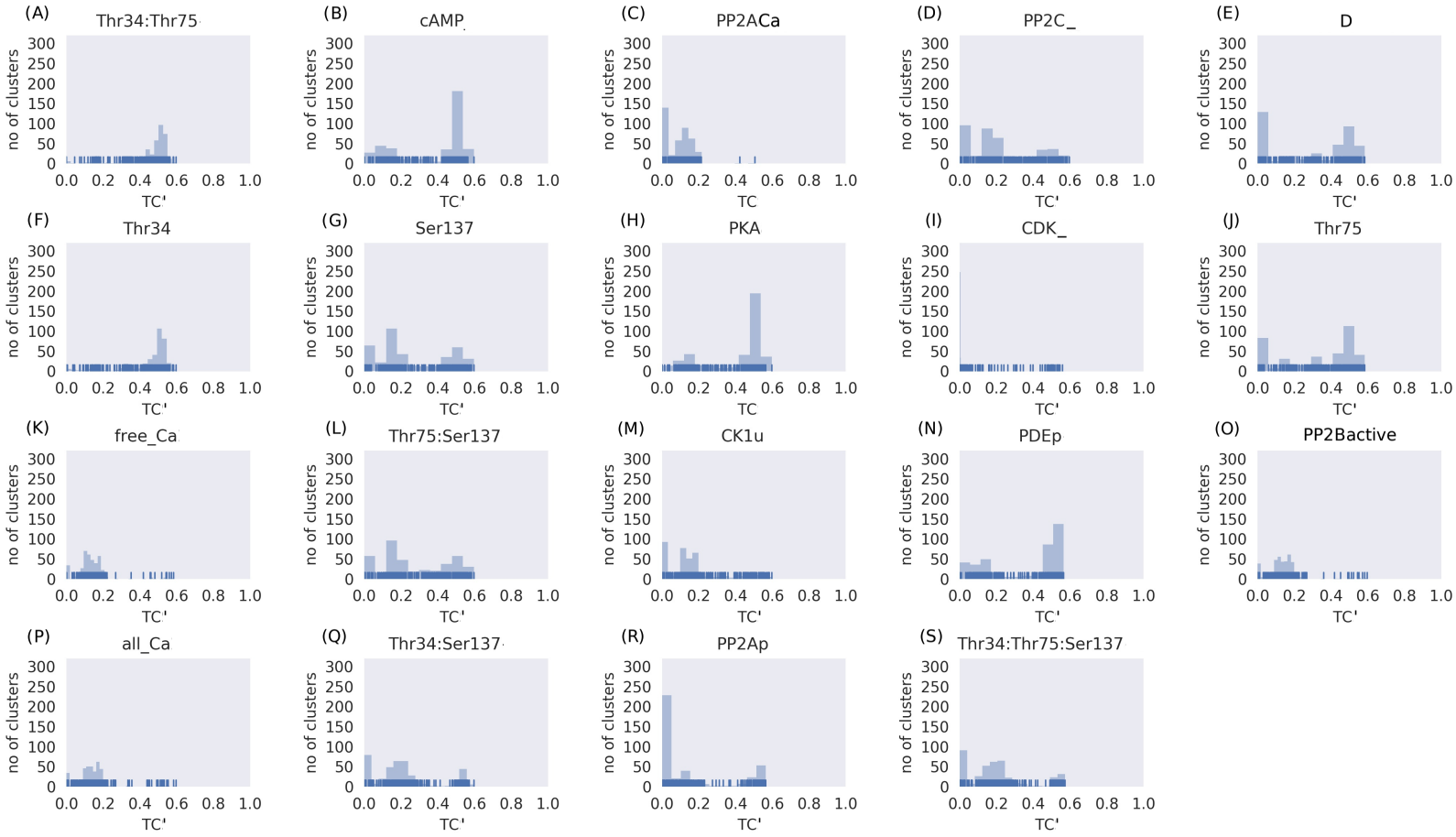


FIGURE 3.22: Distribution of normalised clusters strengths (TC') per observable. Individual data points are displayed as rug plots superimposed on histograms. The normalisation involved dividing cluster strength by cluster member counts if the cluster strength is > 0 and has more than one member. The normalisation has sharpened modalities observed in unnormalised TC values in FIGURE 3.21. Moreover, observables can be grouped by similarity in shape of distributions, such as “Thr34:Thr75” (A) and “Thr34” (F); “cAMP” (B), “PKA” (H) and “PDEp” (N); “Ser137” (G), “Thr75:Ser137” (L), “Thr34:Ser137” (Q) and “Thr34:Thr75:Ser137” (S).

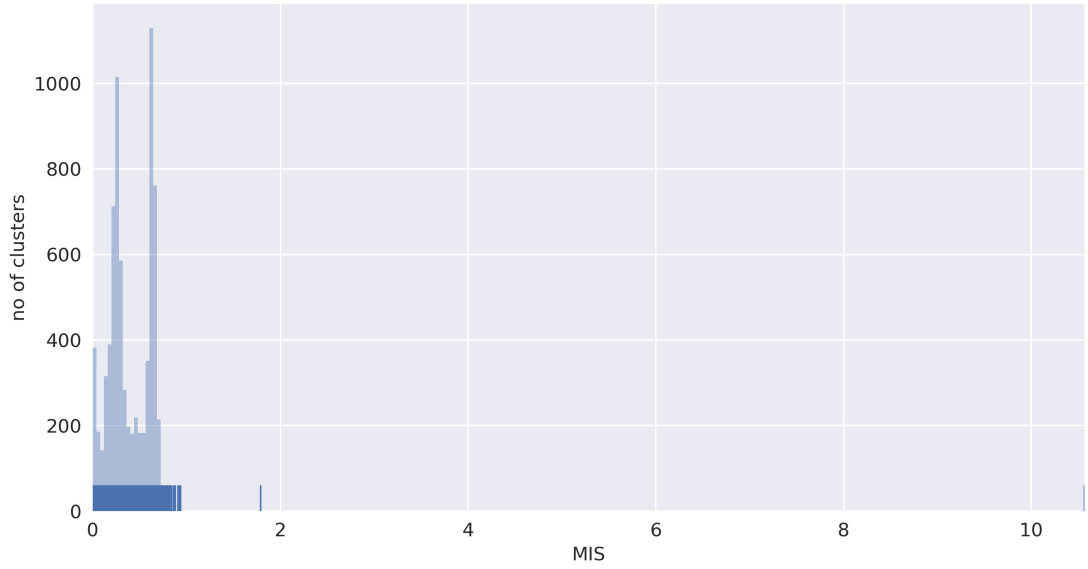
modalities are mainly located around 0.0, 0.2 and 0.5, meaning that despite broad range of cluster strength values to which an observable was classified, there are three most commonly appearing cluster strengths that observables are classified to. Observables can be grouped with respect to shape of distributions such as "Thr34:Thr75" (A) and "Thr34" (F); "cAMP" (B), "PKA" (H) and "PDEp" (N); "Ser137" (G), "Thr75:Ser137" (L), "Thr34:Ser137" (Q) and "Thr34:Thr75:Ser137" (S); "free_Ca" (K) and "all_Ca" (P). These groupings refer to different forms of the same agent (P,K) or directly reacting agents (B, H, N).

Another metric to examine is distribution of observable strengths *MIS*. Figures 3.23 show distribution of *MIS* for all clusters before (Figure 3.23A) and after *MIS* scores for singleton clusters were set to 0 (Figure 3.23B). *MIS* score is generally kept between the range [0,1] but can have an outlier values that exceed this range. After removal of degenerated clusters, *MIS* values are kept within the range. In Figure 3.23B, there are two most frequent scores with means around 0.2 and 0.6.

In Figure 3.24, this distribution is parcelled into observables. Most of the observables take the full range of values from 0.0 to around 0.75. Though the same 7 observables tend to take values around the higher end of the range, the view of *MIS* values per observable shows that other than 7 mentioned observables have relatively higher scores, such as "Thr75:Ser137" (P), "Ser137" (O), "Thr34:Thr75" (Q), that mean their dependence to the cluster they are associated to is strong. There are also observables with high tendency to take values around 0 ("CDK5_", E) or 0.25 ("free_Ca", B; "all_Ca", C; "PP2ACa", H; "PP2Bactive", J).

As seen in *Section 3.4.1*, cluster indices are ordered according to their *TC* scores, locating the strongest one on the index 0. Figure 3.25 shows distribution of cluster indices from 0 to 9 on a raw CorEx output dataset (before any degenerated clusters had their *TC* or *MIS* values setup to 0). A "G-" suffix means a group followed by the cluster index number to which observable was classified. Again, the same 7 observables are discernible, as they are all located in the "G0" cluster in at least half of the total of 400 clusterings (C, D, E, H, K, L). Some other observables are more likely to be located in the "G1" or "G2" clusters ("free_Ca", F; "all_Ca", Q; "PP2ACa", R; "PP2Bactive", P). There is also an observable, "CDK5_" (J), with the opposite tendency to be located in the weakest clusterings (e.g. "G9").

(A)



(B)

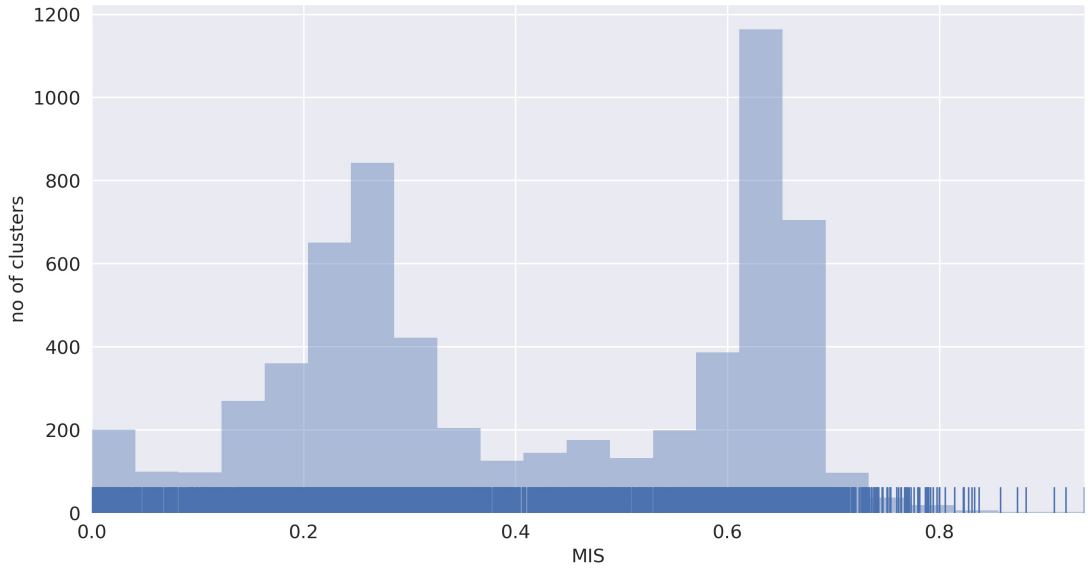


FIGURE 3.23: Distribution of observable strengths (MIS) defining dependence between observables and their clusters, to the number of clusters. Clusters are derived from clusterings of time courses obtained with all parameter samples. In general MIS take values within range of $[0,1]$, however, anomalous values are also encountered (A). By setting MIS values of observables assigned to singleton clusters defined as degenerate ones, the range of MIS values is kept between $[0,1]$.

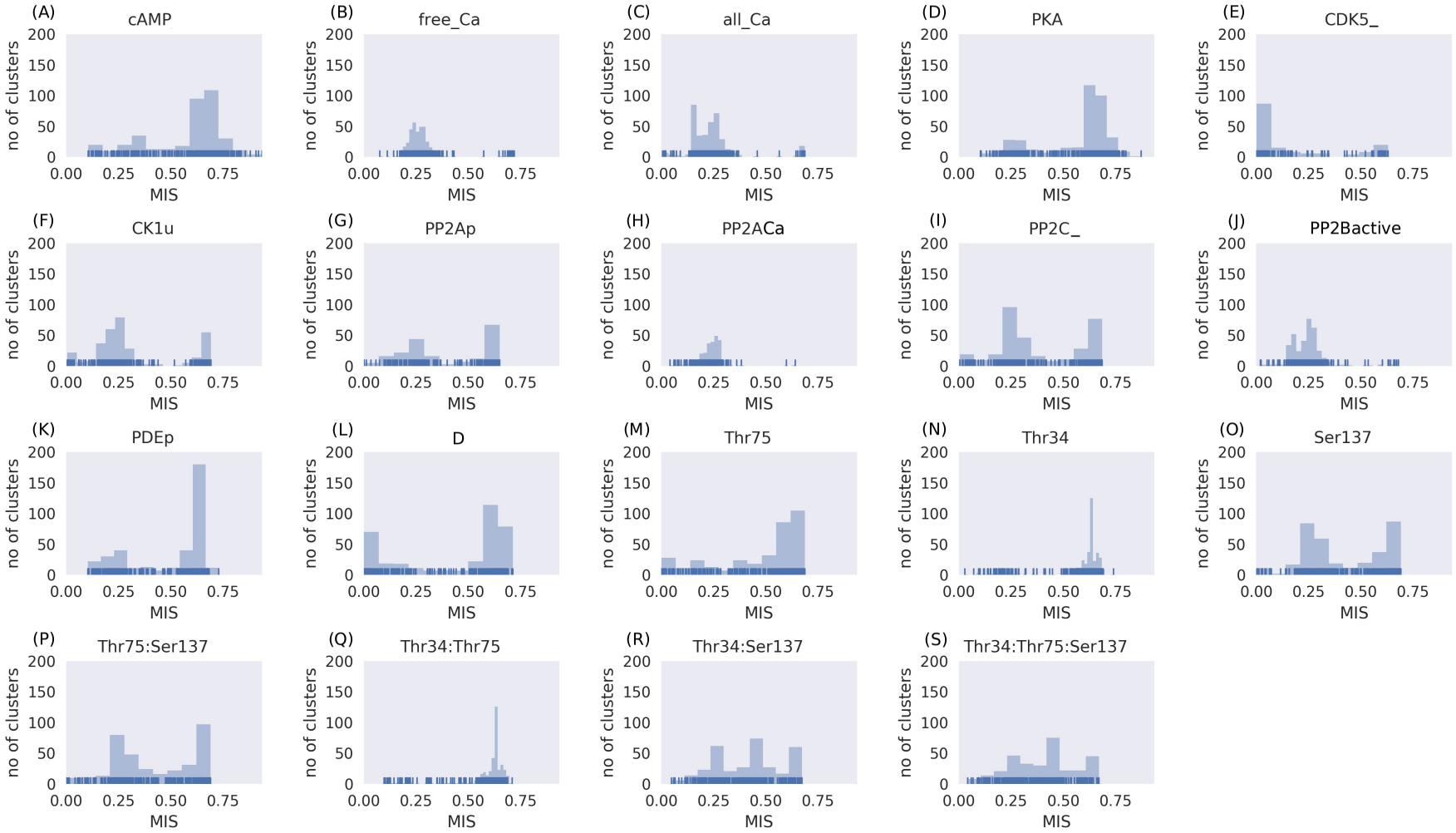


FIGURE 3.24: Distribution of *MIS* for each observable after *MIS* of observables found in singleton clusters were set to 0.

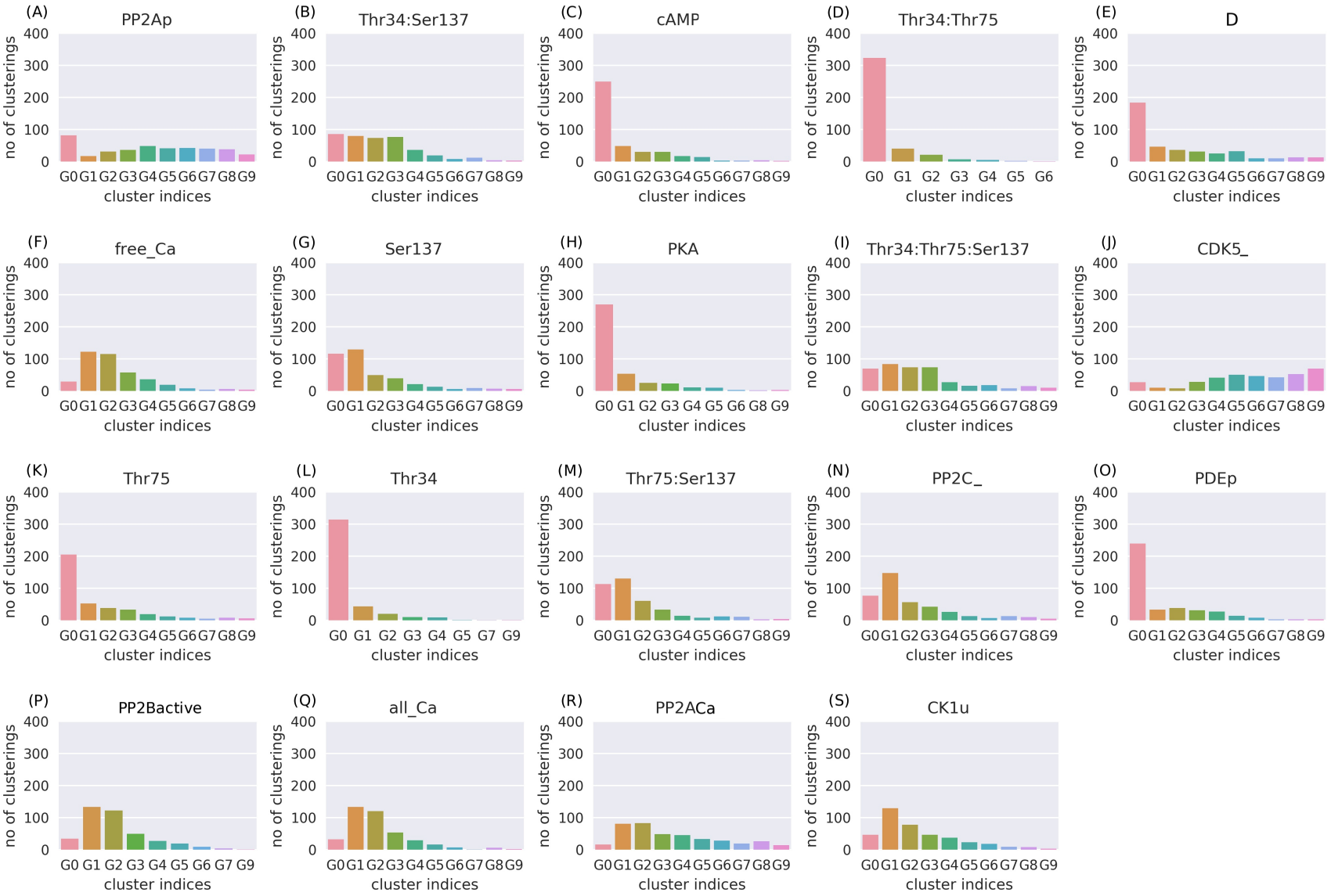
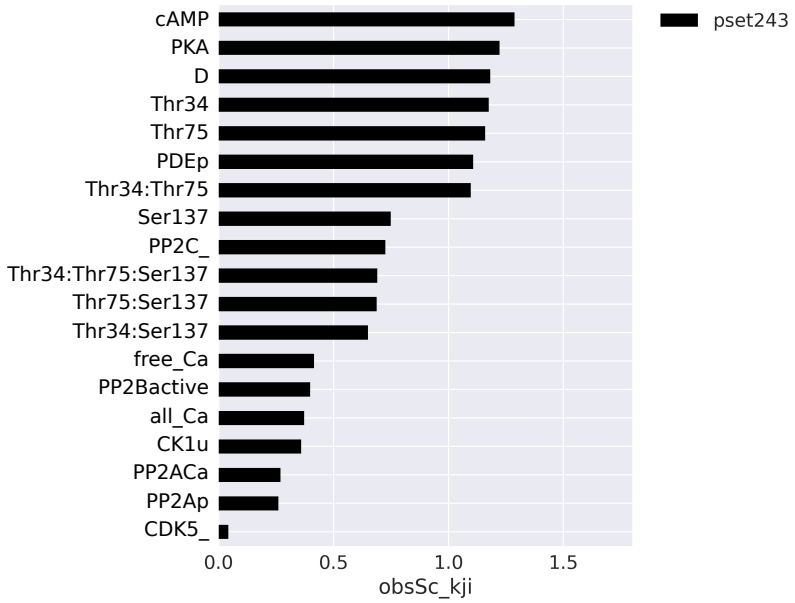


FIGURE 3.25: Allocation tendency of observables to cluster indices across all clusterings (raw data set).

(A)



(B)

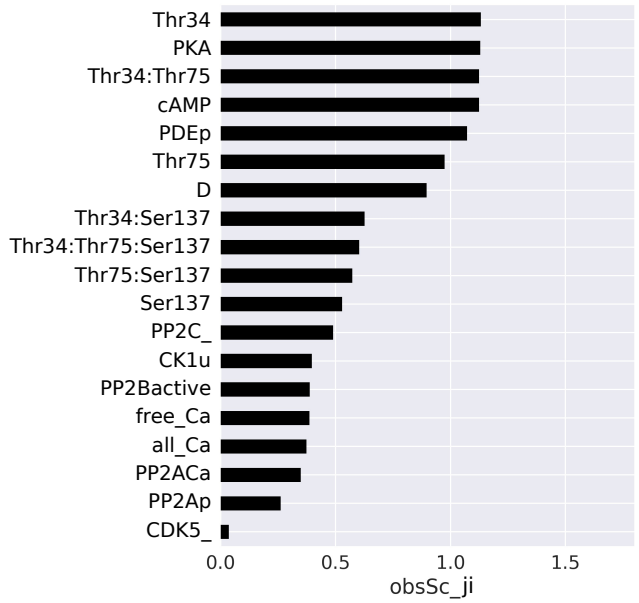


FIGURE 3.26: Comparison of two observable scores calculated from CorEx output measures derived from clustering of 400 averaged time courses obtained from executing the model of DARPP-32 network with varied number of parameter sets. The first observable score, defined in Equation 3.13 and calculated per clustering type, is based on CorEx output measures derived from only one identified clustering type, named "pset243", composed of two clusterings. Being such narrow representation of the whole data set composed of 400 clusterings, the observable score with respect to clustering type reference to the observable score defined in Equation 3.16 that include CorEx output measures from all clusterings. Despite the difference in two observable score definitions, the same 7 observables are enlisted as top ones, though with different order. These 7 observables are found among 9 that were classified to the dominating cluster found in clustering of a single set of time courses generated with the model with the basal parameter set.

At the final step, results of complete calculation of observable scores are presented in FIGURES 3.26. The observable scores calculated with respect to all clustered parameter samples (FIGURE 3.26A) are compared with observable scores calculated per identified clustering type with Equation 3.13 (FIGURE 3.26B). Results of latter scoring are presented just for comparison, as there was only one recurring clustering type ("pset243") composed of 2 clusterings. Both scores indicate the same 7 observables as top scored ones, though in different order ("Thr34:Thr75", "cAMP", "D", "Thr34", "PKA", "Thr75" and "PDEp"). All these observables are directly involved in the cAMP signal. In case of score per clustering type, the 7 observables are located in the same cluster, that has the highest score among the other clusters. For score based on metrics collected from all clusterings, the 7 observables are distinctively separated from the remaining 12. For these 7 observables, parameter sensitivities are calculated and presented in the next section.

3.4.4 Parameter scores

62 parameter sensitivity scores are calculated for each of the 7 selected observables, for each time point between 402 and 1200 interval of the simulation. Based on an example of the "Thr34" observable, FIGURE 3.27 demonstrates results of these calculations. The sensitivity scores take values between [0,0.20]. Shapes of curves formed with sensitivity scores calculated over time reflect three different phases of the simulation. These are the cAMP pulse is introduced in the 200th second of the simulation, followed by the Ca^{2+} spiking at the 250th second that lasts until the 300th second. The third phase is relaxation time that starts after the Ca^{2+} spiking ceases at the 300th second and lasts until the end of the simulation. Based on the identification of these intervals on sensitivity curves that bounced off the baseline level, parameters can be divided into the ones that "Thr34" is sensitive to only at the cAMP pulse (e.g. "k57"), during the Ca^{2+} spiking (e.g. "kon6","kcat6", "kon33", "kon41"), and both at the cAMP pulse and the relaxation time (e.g. "kcat1","kon9","k58"). Decision at which individual points of the simulation to calculate sensitivity scores might be difficult to make as sensitivity scores even differ within these three well defined intervals. For instance, these variations in sensitivity scores are observed for the "k58" parameter during the relaxation phase, and for the "k41" parameter during the Ca^{2+} spiking. Furthermore, consideration of

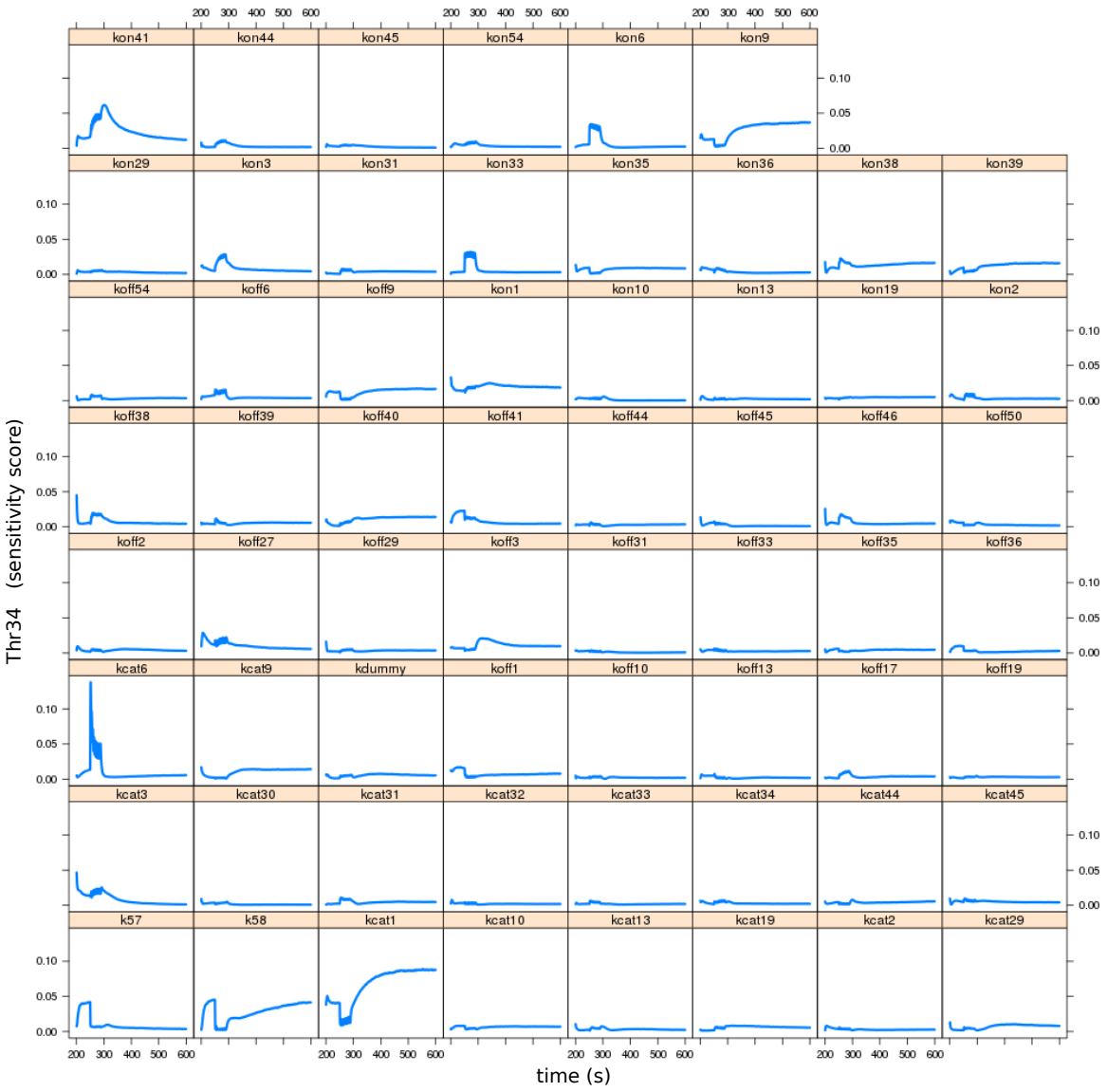


FIGURE 3.27: Sensitivity scores obtained with HSIC-based indices for the "Thr34" observable per each parameter and time step between 201 and 600 second. The sensitivity scores take values between $[0, 0.20]$. At the 200th second of the simulation, the cAMP pulse is introduced. Next, at the 250th second, the Ca^{2+} spiking is initiated that lasts until 300th second. A period passed the 300th second until the end of the simulation defines relaxation time. Shapes of curves formed from sensitivity scores calculated over time reflect patterns of these three different simulation phases. Parameters that sensitivity curves bounce off the baseline divide into ones that "Thr34" is sensitive to only at the cAMP pulse (e.g. "k57"), during the Ca^{2+} spiking (e.g. "kon6","kcat6", "kon33", "kon41"), and both at the cAMP pulse and the relaxation time (e.g. "kcat1","kon9","k58"). Confronted with such diversity of parameter sensitivities conditioned by the time point, it would be difficult to decide for which individual points of the simulation the calculation of sensitivity scores should be performed as sensitivity scores even differ within these three well defined intervals.

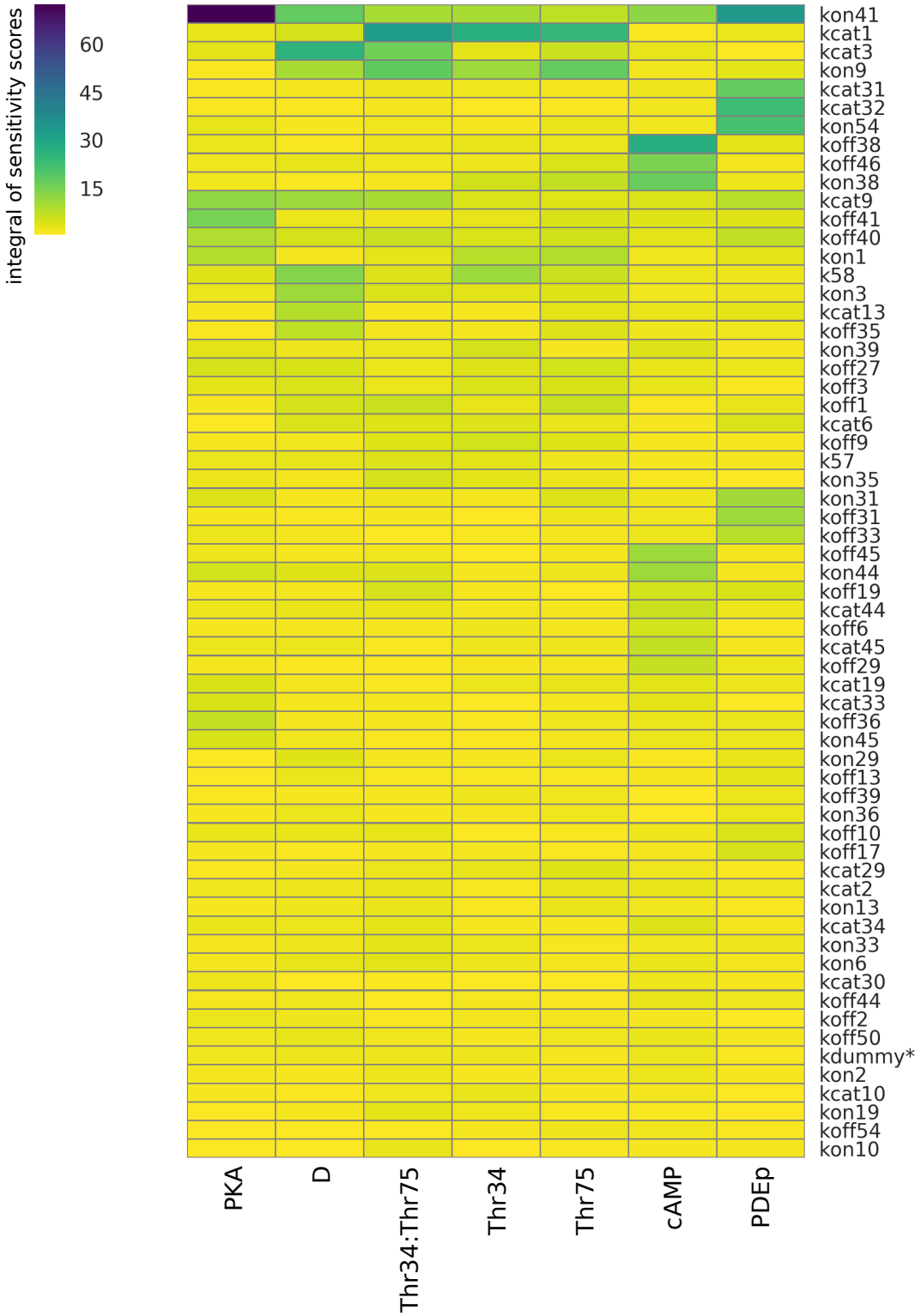


FIGURE 3.28: Clustered heatmap of integrated parameter sensitivity scores for the 7 selected observables. Clustering is performed with Ward's method [296]. Values of the integrals of sensitivity scores are within a broad range of $[0,70]$. Parameters with distinctively higher sensitivity scores divide into parameters affecting multiple observables and parameters that variation only affects particular observables. Of 61 parameters plus a negative control parameter ("`kdummy*`"), around 10 parameters per observable have distinctively elevated sensitivity score exposing more than 80% of parameters as weakly or uninfluential.

these many sensitivity scores for more than one observable poses difficulties in further analysis. Therefore, integral of area under the curve of sensitivity scores define the final parameter scores. By taking the integral of the whole time interval erases the stratification into three different phases of the simulation but determines the overall magnitude of parameter impact. FIGURE 3.28 summarises results of integration of sensitivity scores for the 7 observables as a heatmap clustered with the agglomerative hierarchical method of Ward. This method applies a minimal variance criterion as a clustering objective function [296]. The integrated scores take values from just above 0 to the maximal of around 70. The top four parameters are "kon41","kcat1","kcat3" and "kon9". All four are important parameters for four molecular species of DARPP-32, with the first one, "kon41", having non-negligible impact on all observables. This parameter is responsible for the speed of re-association of PKA to R2C2. Therefore, "PKA" and the phosphorylated phosphodiesterase (PDE) ("PDEp"), as a target of the "PKA" kinase, are most sensitive observables to this parameter. The next parameter, "kcat1", decides on the speed of the phosphorylation of Thr75 by CDK5. Therefore, it is rather straightforward why this parameter has an impact on such observables as "Thr34:Thr75" and "Thr75". The same parameter is also important for the "Thr34" observable as Thr34 cannot be phosphorylated by PKA if it is already phosphorylated on Thr75. Next in the line is "kcat3". It is a constant rate parametrising the phosphorylation of Thr34 by PKA, when Thr75 is unphosphorylated. The highest integral score for this parameter is allocated for the unphosphorylated DARPP-32 ("D"). This might be due to that the phosphorylation of Thr34 mainly occurs on the unphosphorylated DARPP-32 [177]. The second observable sensitive to this parameter is "Thr34:Thr75". Explanation for this importance relation can be seek in the order of reactions, that is the Thr34 site has to be phosphorylated before Thr75 as Thr75 blocks phosphorylation of Thr34 by PKA. The last on the list of the four parameters is "kon9". This is a binding parameter of the unphosphorylated PP2A to DARPP-32. PP2A is the phosphatase of Thr75 and in this way, it is important for all molecular species of DARPP-32 phosphorylated at Thr75.

Next to parameters affecting multiple observables, there are parameter groups highly scored only with respect to a particular observable. For instance, there is a group of three parameters, "kcat31","kcat32","kon54", that are highly scored with respect to "PDEp". The first two parameters are rate

constants of phosphorylation and dephosphorylation reactions of PDE, respectively. A link with the third "kon54" parameter, is less straightforward as it is a rate constant in a binding reaction of Ca^{2+} to PP2A. The reason why "kon54" is important for "PDEp" can be explained in perspective of encoded dependencies in the model. When Ca^{2+} is bound to PP2A, PP2A has a four times lower dissociation rate from DARPP-32. Being the phosphatase of Thr75, PP2A has then a greater chance to dephosphorylate its target. A more important aspect is that PP2A competes with PKA for DARPP-32. When PKA is blocked from binding to DARPP-32 by PP2A, then its availability for binding and phosphorylating PDE is enhanced. Therefore, a connection of the phosphorylated PDE to PP2A is mediated by PKA.

To examine if there is any artefactual level of sensitivities, a negative control parameter was included in the sensitivity computation, "kdummy*". This parameter is absent in the model and therefore, neutral to the output results [262, p. 74]. In theory, the sensitivity of such parameter should be equal zero. However, Marino et al. [268] showed that with PRCC and eFAST methods "dummy parameters" take non-zero sensitivity values. Whether such artefactual sensitivities appear in the HSIC-based indices has not been tested before. Figure 3.28 shows "kdummy*" parameter clustered at the bottom, among least scored parameter. A more detailed view on a range of values that this parameter can take can be seen in Figure 3.29. The figure shows distribution of integrated sensitivity indices for each parameter. Indices are gathered from all 7 observables. Distributions are divided into 1st, 2nd and 3rd quartiles to demonstrate variability in parameter distributions. In an ascendingly ordered dataset, the 1st quartile, equivalent to 25th percentile, is the median of the range of values between the 2nd quartile and the fist value. The 3rd quartile, equivalent to 75th percentile, is the median value of the range between the 2nd quartile and the last value of this dataset. The 2nd quartile is the median of the whole range of dataset values. In Figure 3.29, end sides of a box indicate the 1st and the 3rd quarterlies with a red line denoting the median. Whiskers reach is defined by $1.5 \times$ Interquartile Range (IQR). IQR is a difference between 3th quartile (Q_3) and 1st quartile (Q_1). The value denoted of the left-hand side whisker is defined as $Q1 - 1.5 * IQR$ and the value of the right-hand side whisker as $Q_3 + 1.5 * IQR$. Parameters are sorted according to the 3rd quartile. The median value of "kdummy*" is 2.0 with the maximal value of 2.42. Conse-

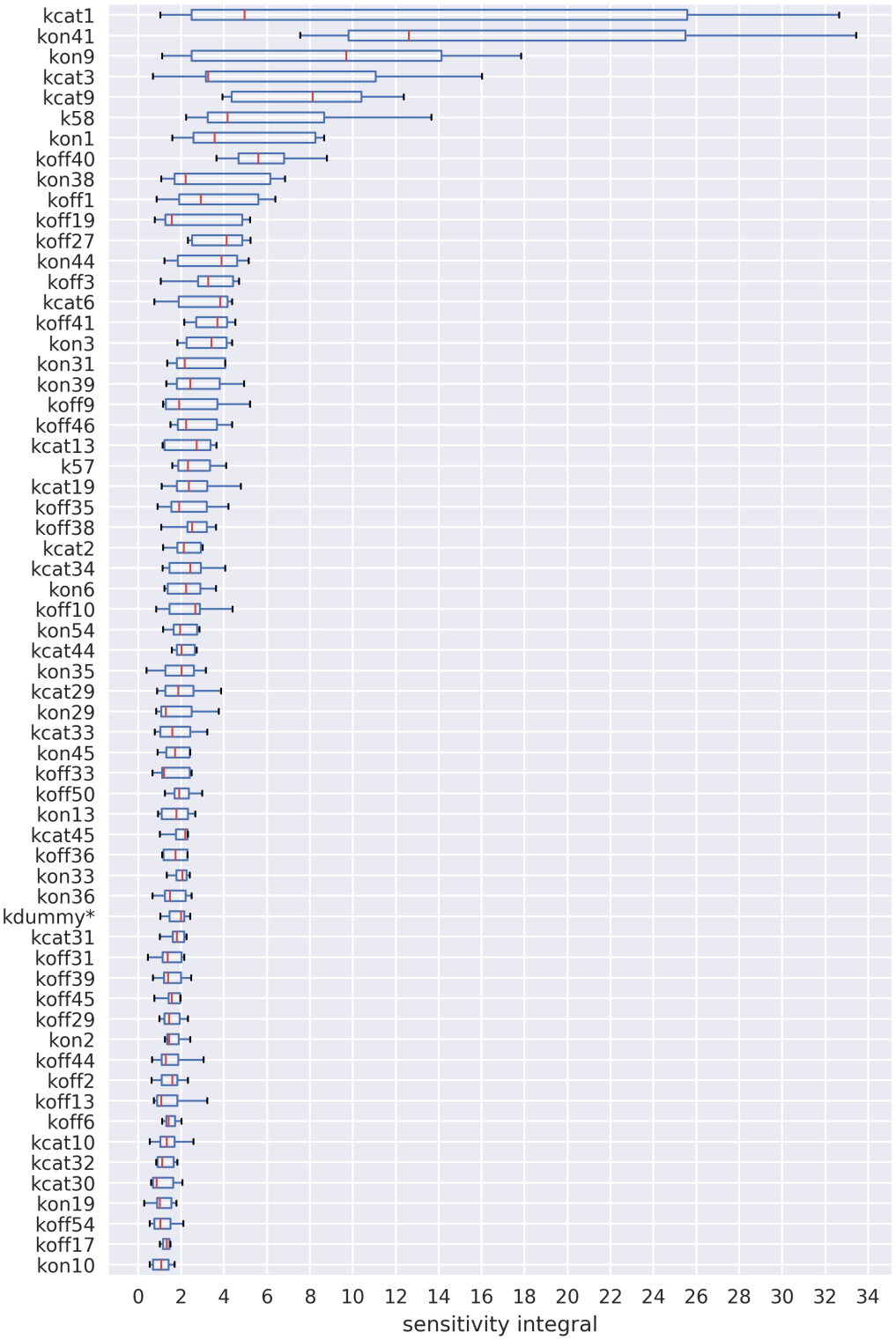


FIGURE 3.29: Distribution of integrated sensitivity scores for each parameter gathered from all observables. Distributions are divided into quartiles to demonstrate scores variations in the wild-type model. End sides of a box indicate 1st and 3rd quartiles with red line denoting the median value. Spread of whiskers is defined with the $1.5\times$Interquartile Range (IQR). Parameters are sorted according the 3rd quartile. Reactions that the top 5 parameters are involved are dephosphorylation and phosphorylation of Thr75 ("kon1","kon9", "kcat9"), phosphorylation of Thr34 ("kcat3") by PKA, and deactivation of PKA ("kon41").

quently, HSIC-based indices are not free from artefactual sensitivities. Though reshuffled, the same four parameters can be identified at the top of the list as in Figure 3.28.

Taken together, highlighted groups of influential parameters have indirect relations to observables they have impact on. Why these key reactions have been emphasised by identified parameters can be well-argued with respect to encoded mechanisms in the model. Rather a small fraction of the total of 61 parameters has distinctive influence per a single observable. Among these, a handful of parameters has significant effect on multiple observables. Lastly, non-zero sensitivity scores gained by negative control parameter indicates that, though with a very low magnitude, HSIC-based sensitivity indices are not free from artefactual sensitivity scores.

3.4.5 Weighted network of observables and parameters

So far analysis of a linearly ordered or clustered lists of parameters were shown. This section presents the alternative view on relations between observables and parameters represented and analysed as a weighted network graph. 62 parameters and 7 observables constitute a fully connected network of 69 nodes and 434 weighted edges. Figure 3.30 shows this network but with a subset of parameters and edges that integral of sensitivity scores reached values above 4 to improve visibility of most affecting parameter sets. The network is composed of 43 parameter nodes, 7 observable nodes, and 94 edges. A common analysis of sensitivity scores discusses the top scored parameters. In this study, though a cut-off of 4 may sound arbitrary, it was chosen to preserve some distance from the maximal sensitivity score acquired by the dummy parameter. Different levels of stringency in selection of the threshold value can be applied that allow for more clear exposition of observed results. Figure 3.31 presents the same network but with more stringent threshold of sensitivity score set to 10. Higher stringency in weights emphasises bridging parameters (“kon41”,“kon41”) between observables representing different forms of DARPP-32 and other 3 observables. Moreover, unphosphorylated DARPP-32 (“D”) is also a bridge between these two observable groups.

In the network graph in Figure 3.30, “PDEp”, “PKA” and “cAMP” observables are located in the corners, connected to parameters important to them alone, and parameters that importance is shared with other observables. The

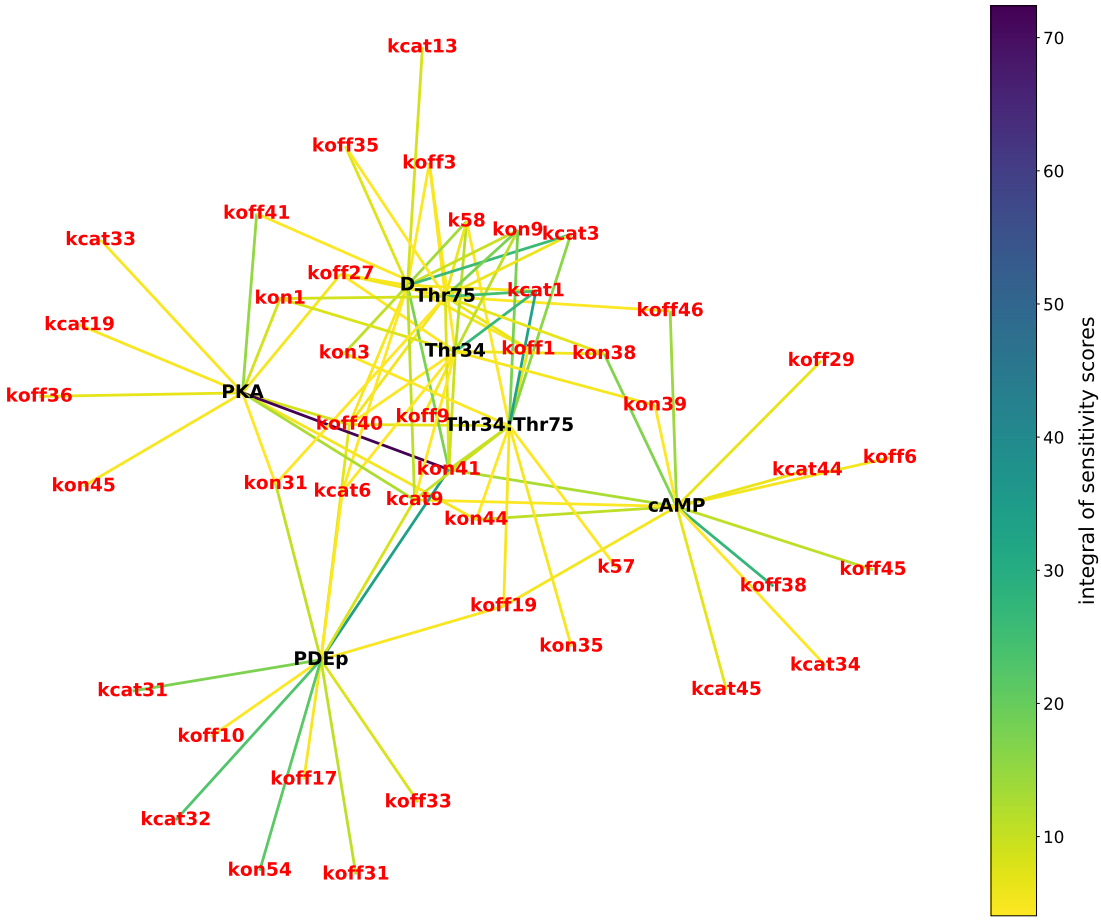


FIGURE 3.30: Network of observables (*black labels*) and parameters (*red labels*) joined with weighted edges. Weights are defined by integrals of sensitivity scores and represented with edge colours with numeric values indicated by the colour map. The network includes parameters that scores reached > 4. The network layout divides it into four regions, a central one where molecular species of DARPP-32 share a great number of parameters with each other and three other peripheral regions, occupied by “PDEp”, “PKA” and “cAMP” observables.

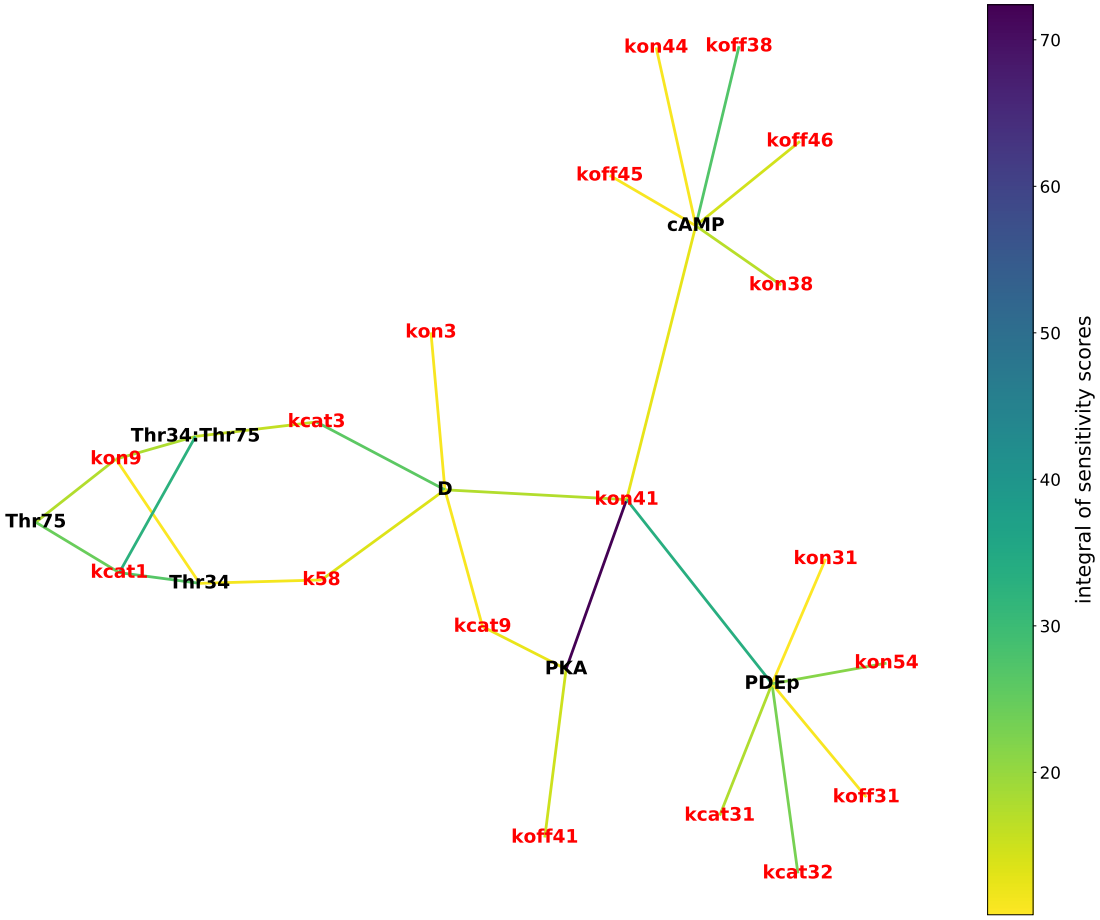


FIGURE 3.31: Network of observables (*black labels*) and parameters (*red labels*) joined with weighted edges. Weights are defined by integrals of sensitivity scores and represented with edge colours with numeric values indicated by the colour map. The network includes parameters that scores reached > 10. This high stringency in parameter scores emphasises bridging parameters ("kon41","kon41") between observables representing different forms of DARPP-32 and other 3 observables. Moreover, unphosphorylated DARPP-32 ("D") is also a bridge between these two observable groups.

central part of the graph is occupied by observables representing 4 molecular species of DARPP-32 that share most of their important parameters. Connections between observables that lead through shared parameters allow to examine relations between observables mediated by connections to the same parameters. By following this observation, one could ask what is the maximally connected subgraph or a network component. This question can be answered by showing the main core of the network graph. Figure 3.32 shows the main core of the network of parameters and observables with the integral sensitivity threshold set to > 4. This subgraph contains nodes of degree ≥ 4, that is a score of the main core for this graph network. This main-core perspective shows the network in four layers, two representing observables and two, parameters. This network preserves only these parameters that have distinctive impact on observables and are shared between at least 4 of all 7 observables. There are 7 such parameters, of which 3 belong to the top 4 parameters identified in the clustered heatmap (Figure 3.28). The other 4 parameters are "`kcat9`", "`koff40`", "`koff27`" in the second layer, and `k58` in the last fourth layer. The parameter defining the speed of dephosphorylation of Thr75 ("`kcat9`") connects 4 observables, among which are 3 observables representing DARPP-32 ("Thr34:Thr75", "Thr34", "D") and a kinase of Thr34 ("PKA"). All 5 observables in this main-core graph are connected with a constant rate of dissociation of cAMP from R2C2 ("`koff40`"). It is one of important reactions leading to the PKA activation by binding of cAMP to R2C2. Other highlighted reactions are dissociation of PP2B, a phosphatase of Thr34, from DARPP-32 when all sites of DARPP-32 are phosphorylated ("`k27`") and Ca^{2+} degradation ("`k58`").

Presented analysis of networks combining parameter and observable scores only exemplify possible perspectives gained with application of different graph-based techniques. This representation might be of an advantage when GSA is performed with respect to multiple model outputs. The example of the 7 observables shows how one can achieve an immediate view on groups of parameters, significant with respect to certain groups of observables, next to parameters and observables that connect separated groups. To take complete advantage of the network representation of observables and parameters, in the next section two different model conditions are contrasted and analysed based on differences in parameter sensitivities.

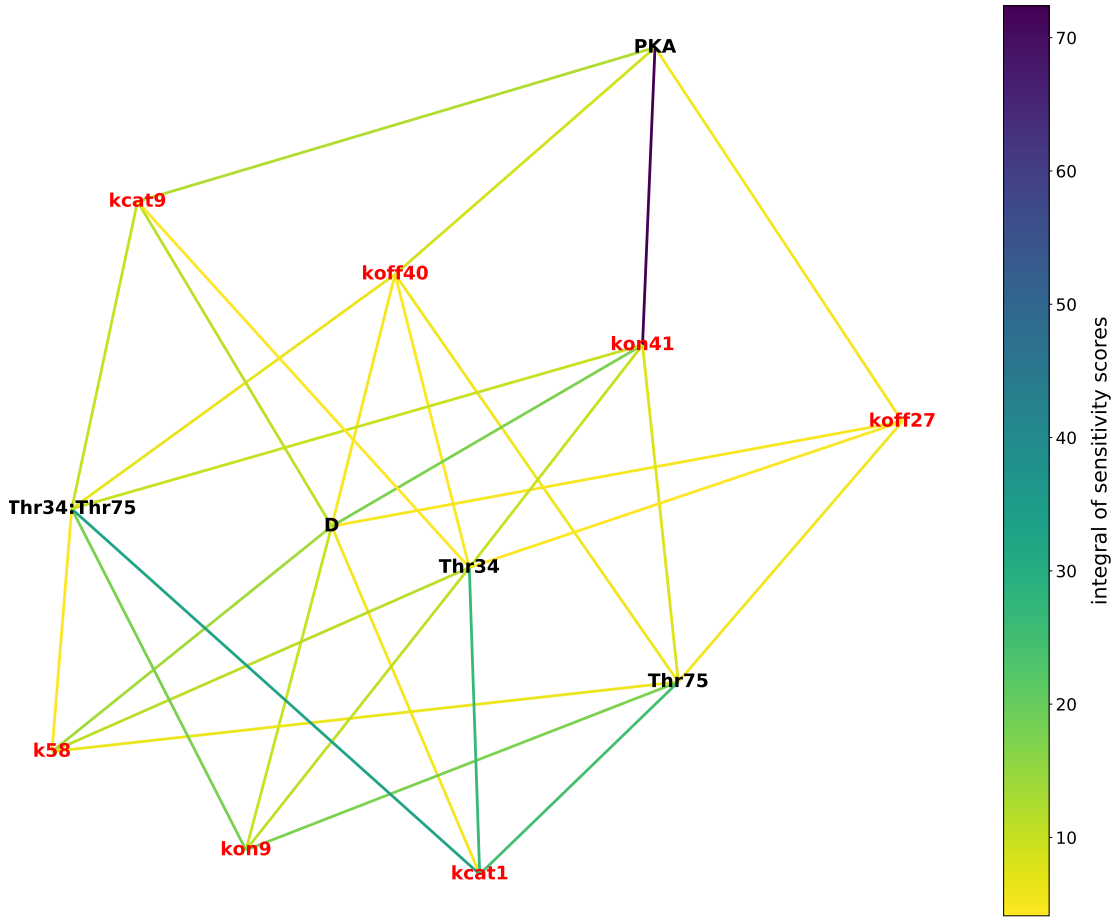


FIGURE 3.32: Main-core of the network of observables (*black labels*) and parameters (*red labels*) joined with weighted edges. Weights are defined by integrals of sensitivity scores mapped to colours as indicated by a colour map and shown as coloured edges. The network includes observables and parameters connected with edges that scores reached > 4 and have degree ≤ 4. This network view preserves only these parameters that have distinctive impact on observables and are shared between at least 4 of all 7 observables.

3.4.6 Studying changes induced by the constitutive Ser137 mutation

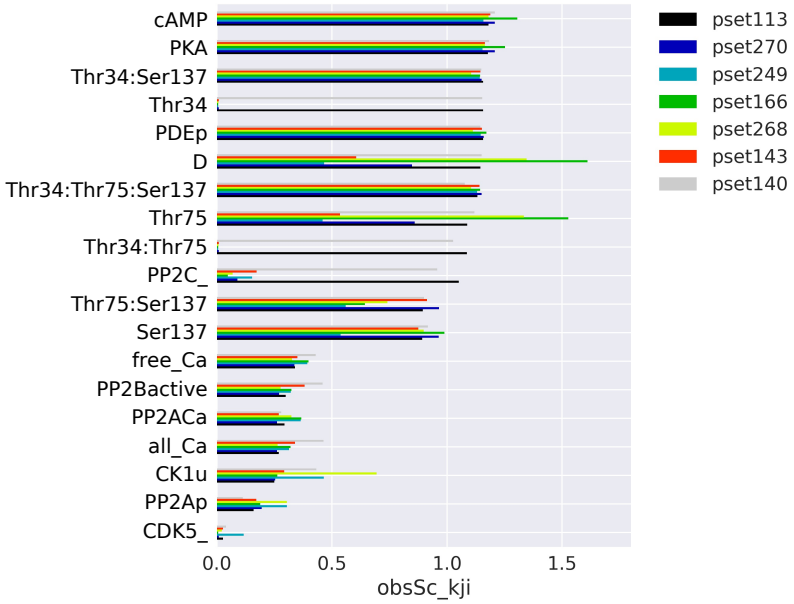
Parameter and observable rankings defined with GSA- and CorEx-derived measures have found legitimate interpretation with respect to mechanisms encoded in the wild-type model. To extend evaluation of these measures and exploit their unified network representation, relations between multiple observables and parameters are studied with respect to different model conditions. The first one, analysed so far, is the wild-type model with the base-line condition. The second is one of two models mimicking site-induced mutagenesis, constSer137. This mutation is designed to induce a sustained phosphorylation of DARPP-32 at Ser137. This mutation is implemented as inactivation of a rule that dephosphorylates DARPP-32 at Ser137. This is achieved by switching the constant rate parametrising this rule to 0 (see *Section 2.3.2.1* for details). The major effect of this mutation is an invertible phosphorylation of the Ser137 site. The phosphorylated Ser137 inhibits dephosphorylation of Thr34 thereby also phosphorylation at the Thr34 site becomes permanent. As the phosphorylation of Thr34 mainly occurs on the unphosphorylated DARPP-32, this completely depletes the level of the unphosphorylated DARPP-32. In consequence, after the cAMP pules leading to the phosphorylation of Thr34, DARPP-32 unphosphorylated at Thr34 and Ser137 is rarely encountered species. Comparison of time courses of the wile-type and constSer137 models can be found in *Section 2.4.2.1* (Figure 2.19. A closer view of observables affected by the mutation can be found in Figure 2.19.

Until now, results of measures obtained with CorEx and GSA were presented only for the wild-type model. Therefore, before the actual comparison between conditions is performed with differential networks, results of measures established for importance of observables, parameters, and their amalgamated representation as networks are briefly demonstrated for the model of constSer137. This will also extend evaluation of consistency and significance of both methods.

3.4.6.1 Observable scores

Observables scores are based on time courses of 19 observables obtained with simulations of the constSer137 model. As in the wild-type model variant, the scores are calculated from CorEx output measures derived from clustering

(A)



(B)

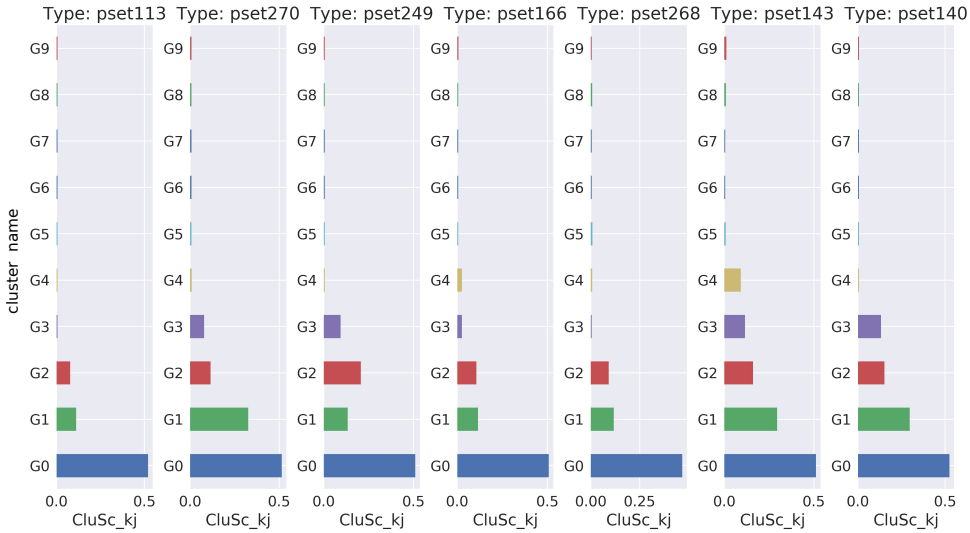
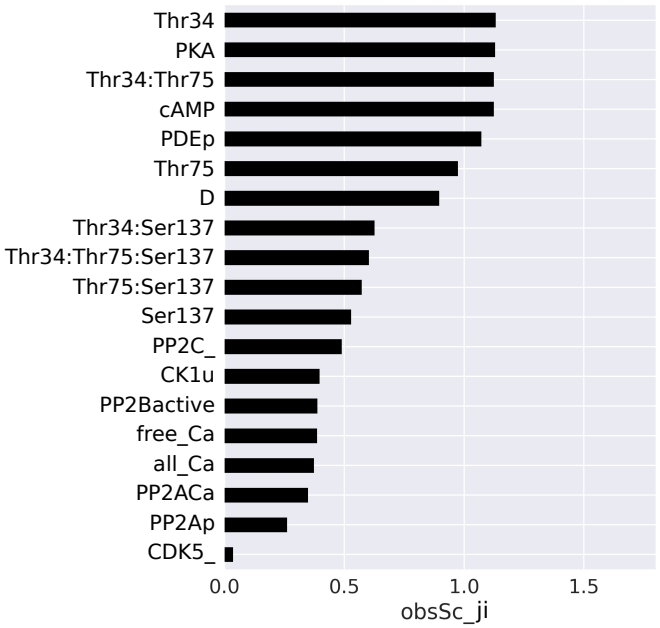


FIGURE 3.33: Observable and cluster scores calculated from CorEx output measures for the constSer137 variant of the DARPP-32 network model. The output measures are derived from clustering of 400 averaged time courses obtained by simulating the model with varied number of parameter sets. The observable score is calculated with Equation 3.13 for each of 7 identified types of clusterings (A). Observables are ordered with respect to scores of the “pset113” clustering type. Variability of observable scores between different clustering types can be observed. In particular, between “pset113” and “pset166”. There are also pairs of clustering types with similarly scored observables, e.g. a pair of “pset113” and “pset140”, and a pair of “pset166” and “pset268”. Comparison of cluster scores for these 7 identified types (Equation 3.15) shows dissimilarity in cluster scores between these pairs (B). As these clustering types are composed of up to 3 clusterings, they constitute a narrow representation of the whole data set of 400 clusterings and therefore, the alternative observable score that include measures from all clusterings is used for observable prioritisation (FIGURE 3.34B).

of 400 averaged time courses resulting from execution of the constSer137 model with randomised parameter sets. Two types of observable scores were defined. The first one calculated per repeated clusterings ($ObsSc_{kji}$) with Equation 3.13. Recurrence of the same partitions of observables into clusters determine a clustering type. Clustering should reappear at least once among 400 clusterings to establish a type. FIGURE 3.33A presents results of this scoring method applied to the constSer137 model. There are more clustering types identified in the constSer137 model than in the wild-type one. The observable scores are separately calculated for each of 7 identified types of clusterings. Despite disparate scores of observables between clustering types, types of clustering with similarly scored observables are identifiable. For instance, a pair of "pset113" and "pset140" clustering types assign hight scores to "Thr34", "Thr34:Thr75", and "PP2C_", that in other types are zero or close to zero. The other two observables that expose variability between clustering types are "D" and "Thr75". Not only scores for these observables distinctively vary over clustering types but also in other two highly similar clustering types ("pset166" and "pset268"), the two observables have distinctively highest scores than the other ones. As each clustering type is defined by time courses obtained with variable parameter sets, the gain of importance by particular subsets of observables over the others is determined by a particular configuration of parameters. Therefore, observable scoring with respect to the clustering type can potentially indicate specific parameter setup that influence the importance of particular observables. Nevertheless, the identified clustering types constitute a narrow representation of the whole data set of 400 clusterings as each clustering type is composed of no more than 3 clusterings. Despite presence of similarities between pairs of clustering types, comparison of cluster scores for these 7 identified types (Equation 3.15) shows that scores allocated per each cluster in these pairs are dissimilar (FIGURE 3.33B). In general, mainly strongest clusters across all clustering types have resembling scores, a tendency already observed with repeated evaluations of CorEx with varied numbers of clusters (*Section 3.4.1*).

Seeing that the number of clusterings partitioned into types is low and to permit for direct comparison between the two model conditions, the alternative observable score ($ObsSc_{ji}$), defined in Equation 3.16, is favoured for observable prioritisation as the same score is used to select observables of the wild-type model. This observable score is calculated with CorEx output mea-

(A) Wild-type



(B) Constitutive Ser137

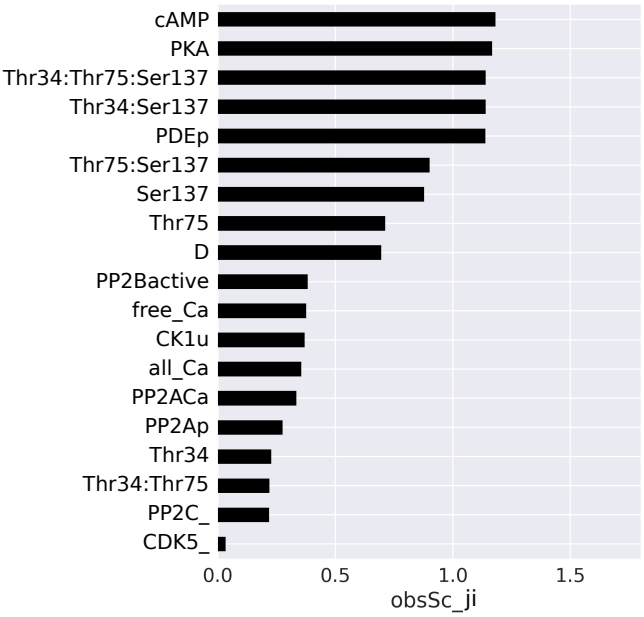


Figure 3.34: Observable scores defined in Equation 3.16 calculated from CorEx output measures for the wild-type model (A) and the constSer137 model (B). The output measures are derived from clustering of 400 averaged time courses obtained by simulating the model with varied number of parameter sets. Differences between modelled conditions are reflected in variable ordering of observables what is found as consistent with the predominant effect of the mutation. For instance, observables representing DARPP-32 phosphorylated at Ser137 scored higher in the constSer137 model rankings compared to the wild-type model. Moreover, the top observable of the wild-type model, representing DARPP-32 phosphorylated only at Thr34, dropped to the 16th position in the constSer137 model rankings.

sures derived from all clusterings disregarding clustering types. FIGURES 3.34 represent a juxtaposition of scored observables with respect to the wild-type model (FIGURE 3.34A) and the constSer137 model (FIGURE 3.34B). Difference between the two models can be observed by comparing orderings of observables between these figures. According to the constSer137 ranking, "Thr34" and "Thr34:Thr75" observables are located at the bottom of the list, at 16th and 17th positions, whereas in the wild-type ranking, these two observables are at the 1st and 3rd positions, respectively. The loss of importance regarding these two observables is consistent with the major effect of irreversible phosphorylation of Ser137. Abundances of molecular species matching observables of DARPP-32 solely phosphorylated at Thr34 ("Thr34"), and phosphorylated at Thr34 and Thr75 ("Thr34:Thr75") are marginal, as they are rapidly phosphorylated at Ser137. For this reason, observables composed of molecular species of DARPP-32 phosphorylated at Ser137 ("Thr34:Thr75:Ser137", "Thr34:Ser137", "Thr75:Ser137" and "Ser137") are among the top 7 highly scored observables in the constSer137 ranking but not among the top 7 of the wild-type list. There are two observables that are located on the boundary in the two rankings. These are "Thr75" and "D", both in the pool of the top 7 observables on the wild-type ranking but dropped below this pool in the constSer137 ranking. They are still located in the upper half of the constSer137 ranking, at 8th and 9th positions (FIGURE 3.34B). In the other type of observable scoring per clustering type, these two observables reach two towering scores in the "pset166" clustering (FIGURE 3.33A). However, compared to others, scores for these two observables are distinctively diverse across clustering types. Both observables stand in contrast to persistently and highly scored observables like "PDEp", "PKA", "cAMP", "Thr34:Thr75:Ser137" and "Thr34:Ser137". Of these 5 observables, "PDEp", "PKA" and "cAMP" are situated at the top of rankings in both modelled conditions.

Based on the observables ranking and in coherence with the wild-type model analysis, the top 7 observables ranked with respect to the constSer137 model are progressed to the next steps of the pipeline. To sum up, among these 7 are observables representing molecular species of DARPP-32 phosphorylated at Ser137, i.e. "Ser137", "Thr34:Ser137", "Thr34:Thr75:Ser137" and "Thr75:Ser137". These 4 observables are absent in the prioritised list of the wild-type model. Among prioritised observables of the wild-type model but

absent in the constSer137 model are "D", "Thr34", "Thr34:Thr75" and "Thr75". Three observables that are analysed in both conditions are "PDEp", "PKA", "cAMP".

It is worth noting that when results of a single run of the constSer137 model in the base-line parameter setup is clustered, these 7 prioritised observables are not located in the same cluster (FIGURE 3.35), contrary to prioritised observables of the wild-type model (FIGURE 3.16). These 7 observables are found in two clusters that consist of 7- and 2-elements. The 2-element cluster with index 2, contains "Thr75:Ser137" and "Ser137" observables. They both have similar and hight observable strengths (MIS_{ji}) signified by edge thickness connecting observables to the cluster (FIGURE 3.35). The 7-element cluster with index 0 that has the other 5 prioritised observables, contains "Thr75" and "D", observables that are located just below the top 7 observables according to the constSer137 ranking.

3.4.6.2 Parameter scores

This section outlines results of calculation of HSIC-based parameter sensitivities with respect to the selected observables of the constSer137 model. The aim is to present key reactions highlighted by top scored parameters. Sensitivity scores are calculated for 61 parameters for each time point between 402 and 1200 interval of the simulation. As the "kcat13" parameter is set to zero to reproduce the constSer137 mutation, it is excluded from calculation of sensitivity scores.

Figure 3.36 shows parameter distributions of integrated sensitivity scores gathered from all observables. Parameters are ordered according the 3rd quantile. Among the first 5 observables, 3 are found in the analogous figure presented for the wild type model (*Figure 3.29*, "kon41", "kon9", "kcat1"). The other 2 parameters of the top 5 of the mutated model, are on the 62nd ("kon10") and 44th ("kon36") positions in the analogous figure for the wild-type model. These two parameters manifest the most prominent effects induced by alteration of the Ser137 function. "Kon10" determines a binding rate of the phosphorylated PP2A to DARPP-32. PP2A binds DARPP-32 to dephosphorylate it at Thr75. "Kon36" defines activation of PP2B that occurs through binding of two Ca^{2+} ions at a single reaction step. PP2B has two roles in the model. It catalyses dephosphorylation of Thr34 and activates CK1, that when

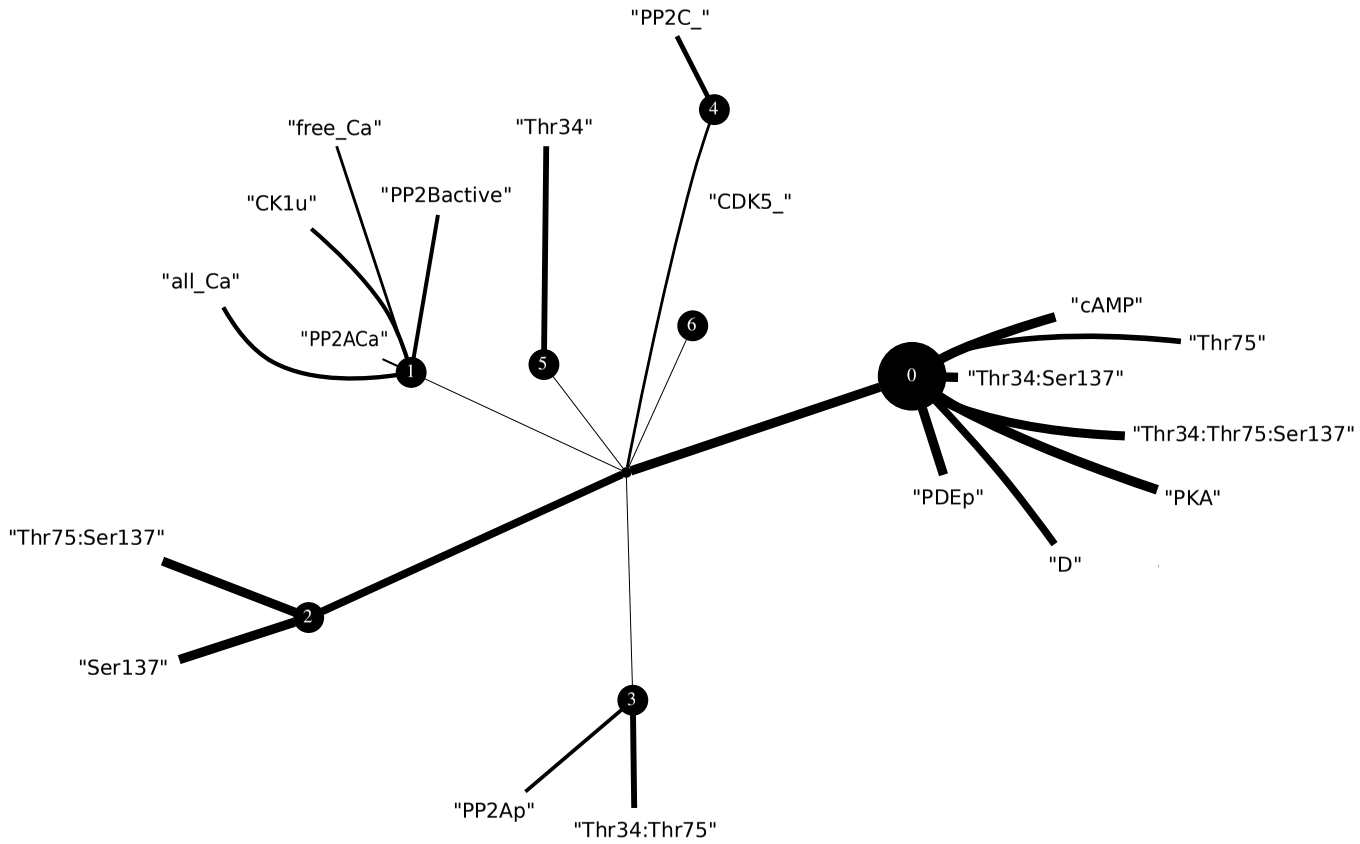


FIGURE 3.35: Tree graph visualising CorEx clustering result for a single of time courses of 19 observables obtained with the model representing constitutive Ser137 phosphorylation. Nodes stand for clusters. Names of cluster members are indicated on ends of outgoing edges. Cluster indices are marked with integers on each node. Node size is proportional to cluster strength. The 0th-indexed cluster is the strongest and the largest one. When the content of this dominating cluster is compared to the content of its equivalent in clustering of the wild-type model (FIGURE 3.16), two absent observables in the cluster of the constSer137 cluster can be noticed. These two missing observables represent DARPP-32 phosphorylated only at Thr34 ("Thr34") and DARPP-32 phosphorylated at Thr34 and Thr75 ("Thr34:Thr75"). Lost position of these two observables is consistent with the major effect of irreversible phosphorylation of Ser137 as molecular species matching the two missing observables are rapidly phosphorylated at Ser137.

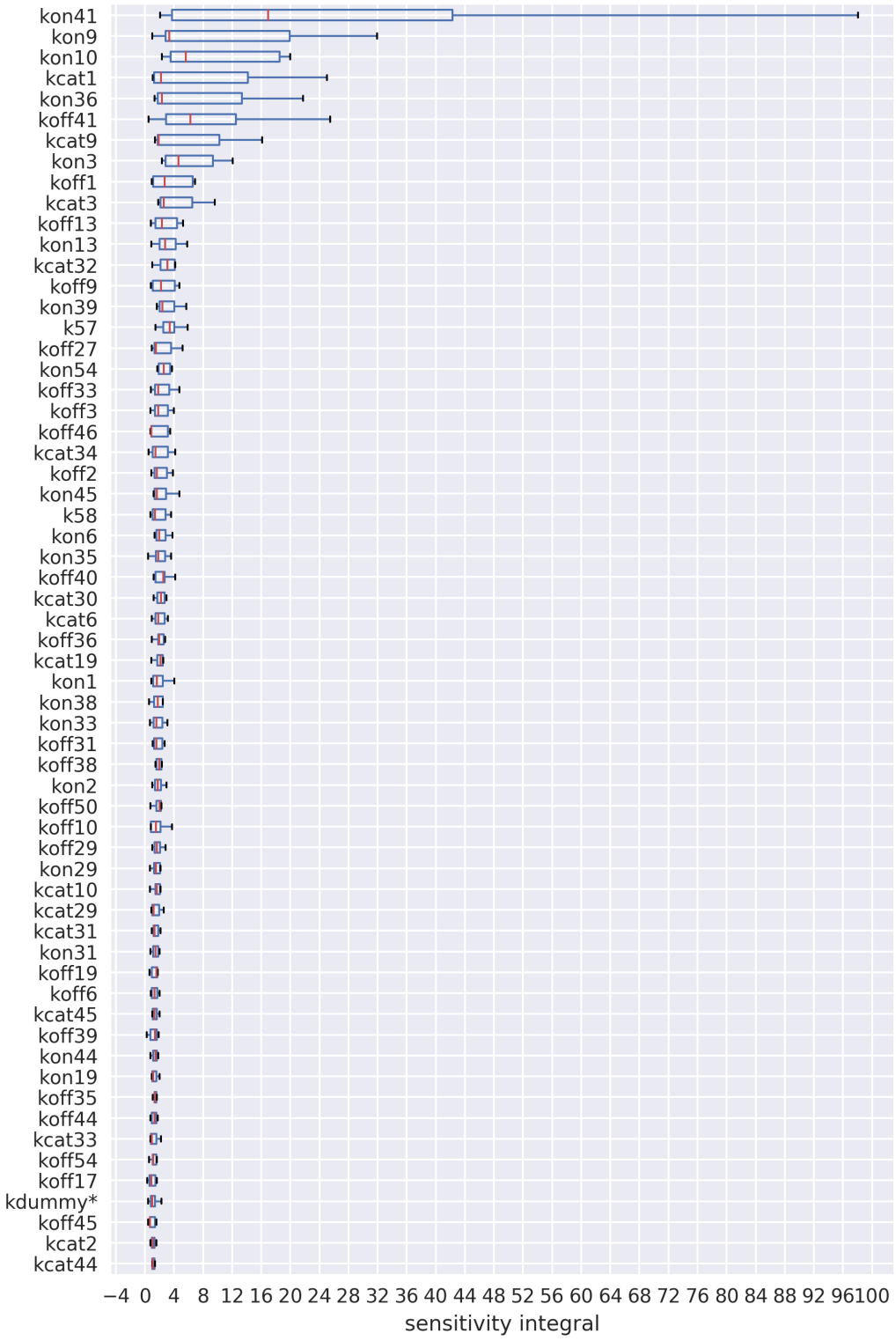


FIGURE 3.36: Distribution of integrated sensitivity scores for each parameter gathered from all 7 observables. Distributions are divided into quartiles demonstrating scores variations in the constSer137 model. End sides of boxes indicate first and third quartiles with red line denoting the median value. Spread of whiskers is defined with the $1.5\times$Interquartile Range (IQR). Parameters are sorted according the 3rd quartile. The top 5 parameters are involved in dephosphorylation and phosphorylation of Thr75 ("kcat1","kon9", "kon10"), activation of PP2B ("kon36"), and deactivation of PKA ("kon41").

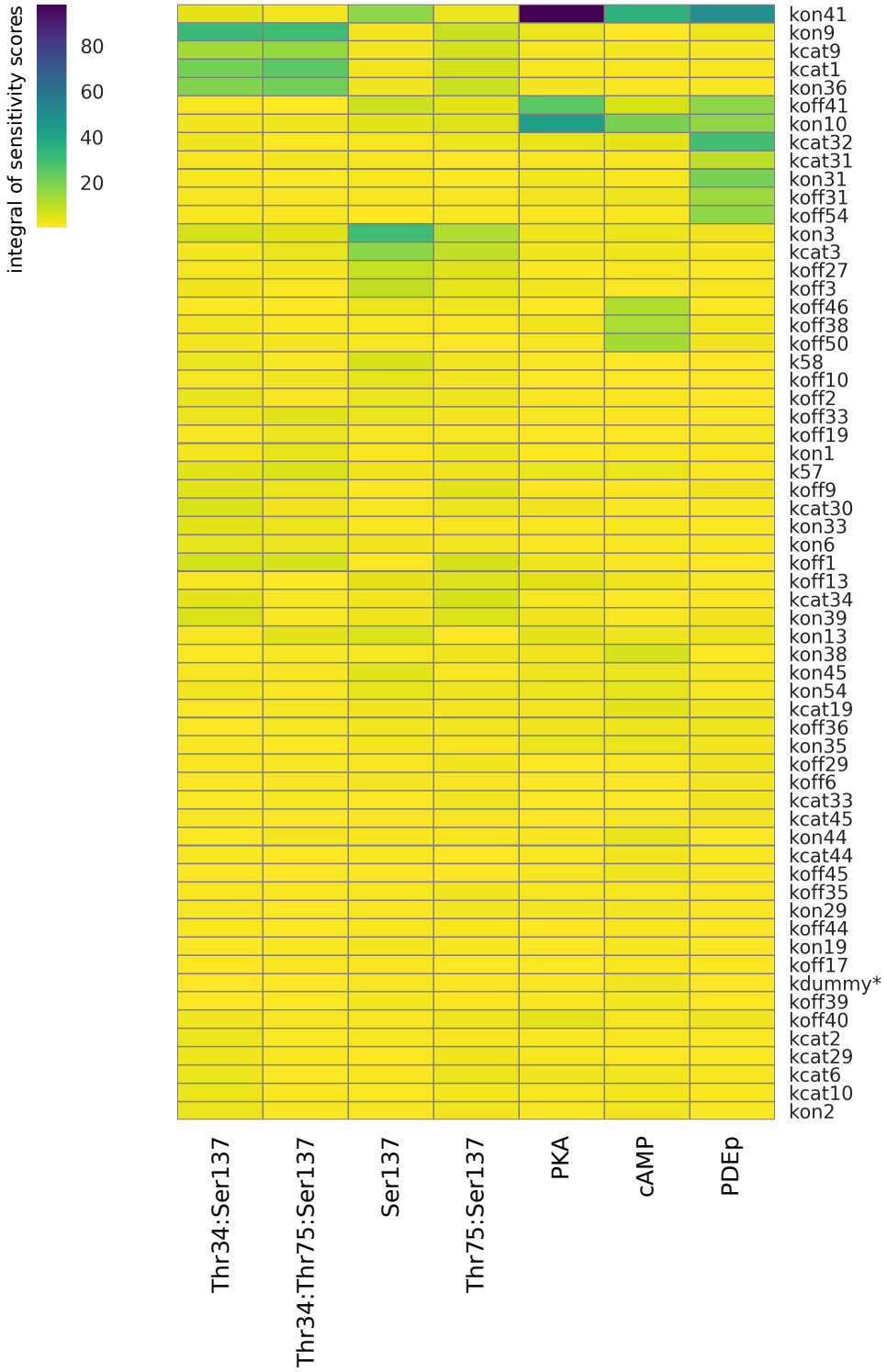


FIGURE 3.37: Clustered heatmap of integrated parameter sensitivity scores for the 7 selected observables of the constSer137 model. Clustering is performed with Ward's method [296]. Values of the integrals of sensitivity scores are within a range of $[0, 100]$. A handful of parameters with distinctively higher sensitivity scores divide into parameters affecting multiple observables, and parameters that variation only affect a single observable. Similarly to parameter sensitivities of the wild-type model, around 10 parameters per observable have distinctively elevated sensitivity scores exposing more than 80% of parameters as weakly or unimportant.

activated, catalyses phosphorylation of Ser137. Increase of importance of this parameter is in agreement with the increase of importance of molecular species of DARPP-32 phosphorylated at Ser137 in the mutated model. Examination of relations between parameter sensitivities and particular observables can clarify importance of these parameters. *Figure 3.37* shows integrated sensitivity scores for the 7 observables as a heatmap clustered with the method of Ward. Figure 3.28 is a corresponding figure for the wild-type model. Integral of sensitivity scores of "kon10" is highly elevated for "PDEp", "cAMP" and "PKA". Non of these observables take part in the reaction parametrised by "kon10", where the phosphorylated PP2A binds DARPP-32. However, a link joining this particular reaction with these three observables is PKA, a binding parter of PP2A and DARPP-32. It is a sufficient tie as "PDEp", "cAMP" and "PKA" observables bind themselves. Moreover, all these three observables directly interact with each other forming a negative feedback loop regulating levels of cAMP (Figure 3.16). cAMP is introduced in the simulation as a large and steeply increased pulse that activates PKA. Elimination of cAMP from the simulation is catalysed by the phoshophorylated PDE. The phosphorylation of PDE is catalysed by PKA. As defined in the rule specification, when PDE is phosphorylated, then the constant rate of binding, unbinding and deactivation of cAMP is doubled. PKA binds to PP2A to catalyse its phosphorylation that increases binding likelihood of PP2A and DARPP-32. As binding between all these proteins is mediated by a single site, PKA is then less likely to bind to PP2A or DARPP-32, what enhances its availability to bind and phosphorylate PDE. These is a setup that is also in force in the wild-type model but "kon10" has not been detected as influential. What causes importance of this particular parameter in the mutation of the Ser137 site? As in the constSer137 model the phosphorylation of Thr34 is close to permanent, PKA is effectively blocked from biding to DARPP-32 when it is phosphorylated at Thr34 due to a general assumption of no product rebinding. Therefore, the link with this particular reaction involving DARPP-32 is rather related to PP2A, at least in the later stages when Thr34 is already phosphorylated. Not being able to bind permanently phosphorylated Thr34, PKA becomes more likely to bind other interactors. These are PP2A, cAMP and PDE. In this way, not only PDE, but also PP2A is more likely to be phosphorylated. "kon10" parametrises reaction where the phosphorylated PP2A binds to DARPP-32, phosphorylated at Thr75. This

phosphorylated form of PP2A is particularly important for "PDEp", "cAMP" and "PKA", as the rate constant parametrising analogous reaction but with the unphosphorylated PP2A ("kon9") is a parameter of distinctive impact only for observables representing various forms of DARPP-32 in both modelled conditions, but not for these three considered observables. Lastly, since all three observables are tracked in unspecified binding state, each of these observables contains trajectories of molecular species that are complexes. For instance, the "PKA" observable is a sum of time courses of all molecular species containing PKA. When PKA is a part of a complex it is less likely to be re-associated to R2C2. In fact a parameter determining the speed of re-association of PKA to R2C2 ("kon41") appears as the most important for all three observables regardless the model variant.

The second parameter that was exposed as particularly influential in the constSer137 model is "kon36". Its sensitivity scores are raised for "Thr34:Ser137" and "Thr34:Thr75:Ser137". Its impact on these particular observables has a clear connection with direct outcome of reaction parametrised by "kon36" as DARPP-32 would not be phosphorylated at Ser137 without activation of PP2B. These two observables are also sensitive to the same most influential parameters that are also among the top 5 ("kon9", "kcat9", "kcat1"). All three parametrise reactions involving the state of DARPP-32 at the Thr75 site. Importance of reactions deciding on this particular phoshorylation site can be linked to dependence between Thr75 and Thr34, the last one being phosphorylated in both considered observables ("Thr34:Ser137", "Thr34:Thr75:Ser137"). This dependence determine that Thr34 is blocked from phosphorylation when Thr75 is phosphorylated. Therefore, there is a certain order how these sites can be phosphorylatated to produce "Thr34:Ser137" and "Thr34:Thr75:Ser137". The same 4 parameters ("kon36","kon9", "kcat9", "kcat1") have elevated integrated sensitivity scores with respect to "Thr75:Ser137". Compared to "Thr34:Ser137" and "Thr34:Thr75:Ser137", these scores are quite low that emphasise importance of these parameters in relation to the phosphorylated Thr34. Interestingly, variation of the same 4 parameters is insignificant for the fourth observable representing DARPP-32 phosphorylated at Ser137 alone ("Ser137"). Two most important parameters for "Ser137" are only shared with "Thr75:Ser137". The first one determines how often PKA binds DARPP-32 when it unphosphorylated at Thr34 ("kon3"), and second, parametrises phosphorylation of Thr34

when DARPP-32 is unphosphorylated at Thr34 and Thr75 ("kcat3").

3.4.6.3 Differential networks of contitutive Ser137 and wild-type models

Analyses of what observables and parameters are important in the constSer137 model have been based on ranked lists so far. Measures collected from results of CorEx and GSA are used to construct a weighted network of parameters and observables for the constSer137 model (Figure 3.38), comparable to the wild type network (Figure 3.30). After preserving only the edges that integral of sensitivity scores reached values above 4 to improve visibility on most affected parameter sets, the network of the perturbed model is composed of 35 parameter nodes, 7 observable nodes, and 72 edges. The same cut-off applied to the corresponding wild-type network yields a slightly smaller network of 43 parameter nodes, the same number of observable nodes, and 94 edges.

To analyse parameters and observables that gained and lost importance due to invertible phosphorylation of Ser137, edge weights of the wild-type network are subtracted from the edge weights of the constSer137 network. As both networks contained different subsets of observables, missing observables were added and connected to all parameters with edges of zero-weights. The procedure of weight subtraction yields a difference network with positive, negative and zero weights. To facilitate analysis of this difference network, it is divided into two networks with respect to edge weights. The first network has only positive edges exhibiting parameters that gained importance in the mutated model, further called the gained-importance network. This network is shown in Figure 3.39A. To increase clarity of the visualisation, only edges with difference in weight values above 4 are drawn. The second network has only negative edges exposing parameters that lost importance due to the perturbation, further called the lost-importance network. This network is visualised in Figure 3.39B. Similarly to the gained-importance network, only edges with weights below -4 are shown. This network has slightly larger (nodes: 43, edges: 74) than the gained-importance network (nodes: 35, edges: 60). This can be explained by the smaller size of the constSer137 network compared to the wild-type model.

The gained-importance network has all 7 observables indicated as important for the mutated model (Figure 3.38). In analogy, the network of lost importance has all 7 observables indicated as important for the wild-

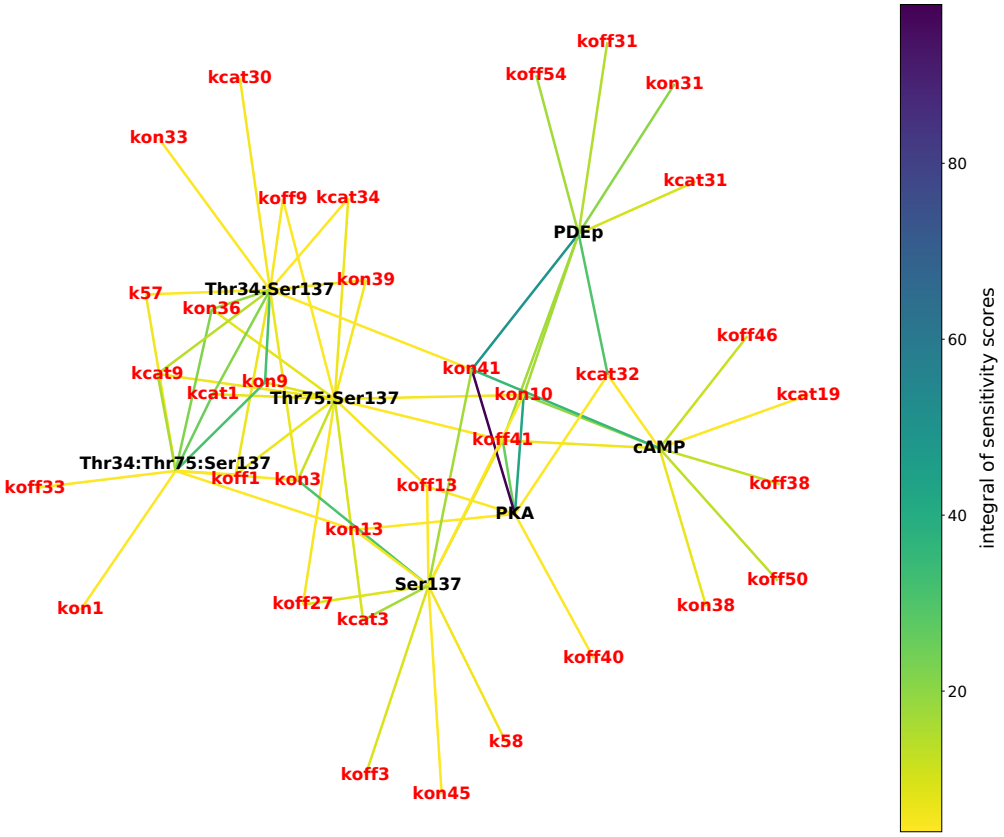
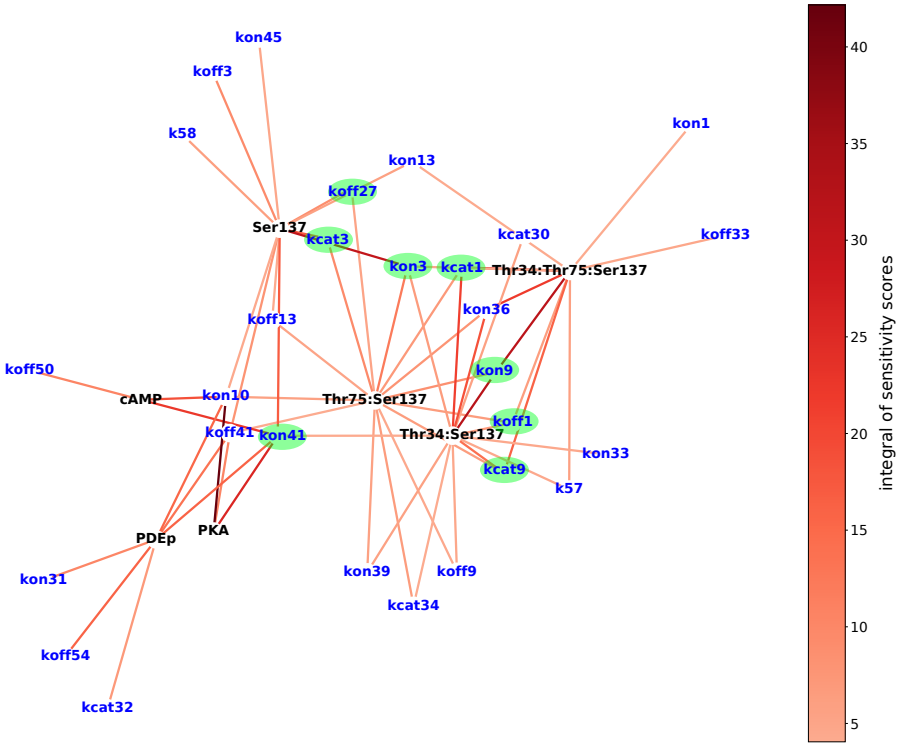


FIGURE 3.38: Network of observables (*black labels*) and parameters (*red labels*) joined with weighted edges for the constitutive Ser137 model variant. Weights are defined by integrals of sensitivity scores and represented with edge colours with numeric values indicated by the colour map. This network plot includes parameters that integral of sensitivity scores is > 4. Parameters can be divided into ones that affect multiple observables and ones that have distinctive impact on a single observable. Analogous to the wild-type model, the network layout is partitioned into two regions that can be determined by the number of connections between observables. The first one allocated to molecular species of DARPP-32 phosphorylated at Ser137 and the other occupied by “PDEp”, “PKA” and “cAMP” observables.

(A) Gained importance



(B) Lost importance

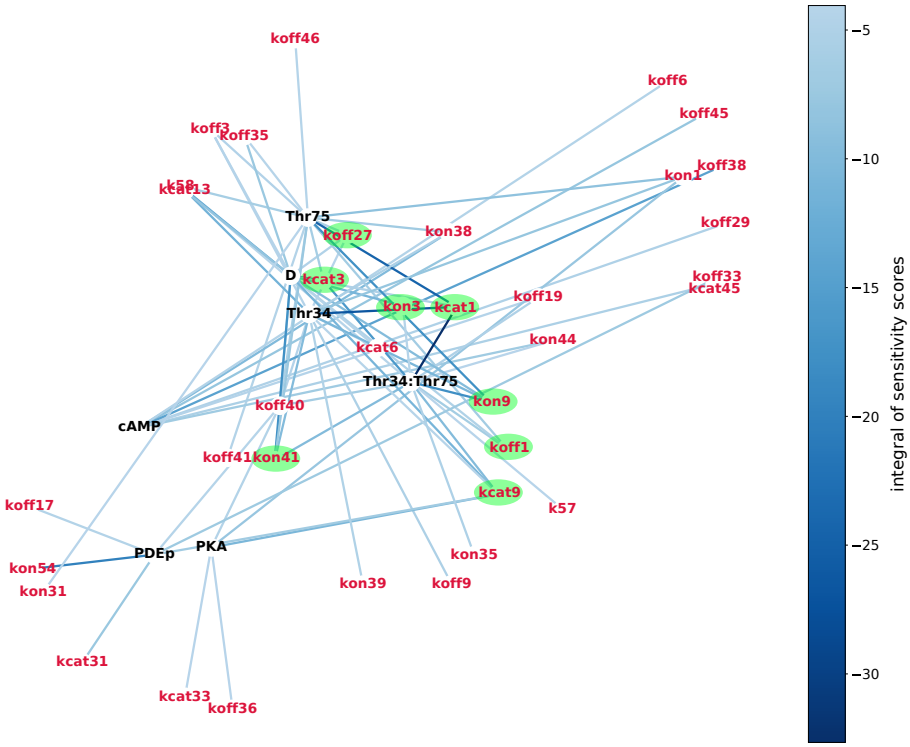


FIGURE 3.39: Differential networks of observables and parameters between the constSer137 and the wild-type model that (A) gained importance (edges with weights > 4) (B) lost importance (edges with weights < -4) with respect to the model with mutation at Ser137. Marked in green are parameters shared between observables representing different state configurations of DARPP-32 regardless the model phenotype.

type model (Figure 3.30). In the gained-importance network, a group of parameters that connects 4 observables only present in the mutated network ("Ser137", "Thr34:Thr75:Ser137", "Thr34:Ser137", "Thr75:Ser137") is the same as in the network of the mutated model (Figure 3.38), as edge weights between these observables and parameters are 0 in the wild-type network. As anticipated, similar holds for the network of lost importance for the 4 observables present only the wild-type network ("D", "Thr34:Thr75", "Thr34", "Thr75") (Figure 3.30). In the gained-importance network, there are clearly three parameters ("kon41", "koff41","kon10") that are connecting observables representing 4 various forms of DARPP-32 to observables forming a negative feedback loop ("cAMP", "PDEp", "PKA"). Comparison of parameters shared between these 4 observables representing DARPP-32 shows that the same 8 parameters can be identified in both networks of gained and lost importance (Figure 3.39). Despite of difference in composition of phosphorylation sites of DARPP-32 and regardless the model phenotype, 8 parameters, that constitute around half of all shared parameters in both networks, are the same. Among these ones, most prominent are: "kon3","kcat3","kcat1", "kon41", "kcat9" and "kon9". Meaning of these parameters was elaborated earlier in analyses of heatmap plots in the context of the wild-type and the constSer137 models (*Section 3.4.6.2*). More interesting are changes of parameters among observables shared by both conditions that form the negative feedback loop, i.e. "PDEp", "PKAa" and "cAMP".

Figure 3.40 shows adjacent parameters to each of the three observables in perspective of three different types of networks. The first type is the wild-type network (Figure 3.40A), and the two others show parameters that gained (Figure 3.40B) and lost (Figure 3.40C) importance in the constSer137 model. For all three observables, the number of neighbours in the wild-type network is higher then the total number of neighbours in both negative and positive networks. It is because among parameters in the wild-type network are ones that difference in sensitivity scores between two conditions is below 4 in the positive networks, and above -4 in the negative networks. Adjacent parameters to all three observables in the networks of parameters that lost importance in the constSer137 condition (Figure 3.40C) are contained in the set of parameters found in the general wild-type network for each of three observables. This is different in the network of parameters that gained importance for respective

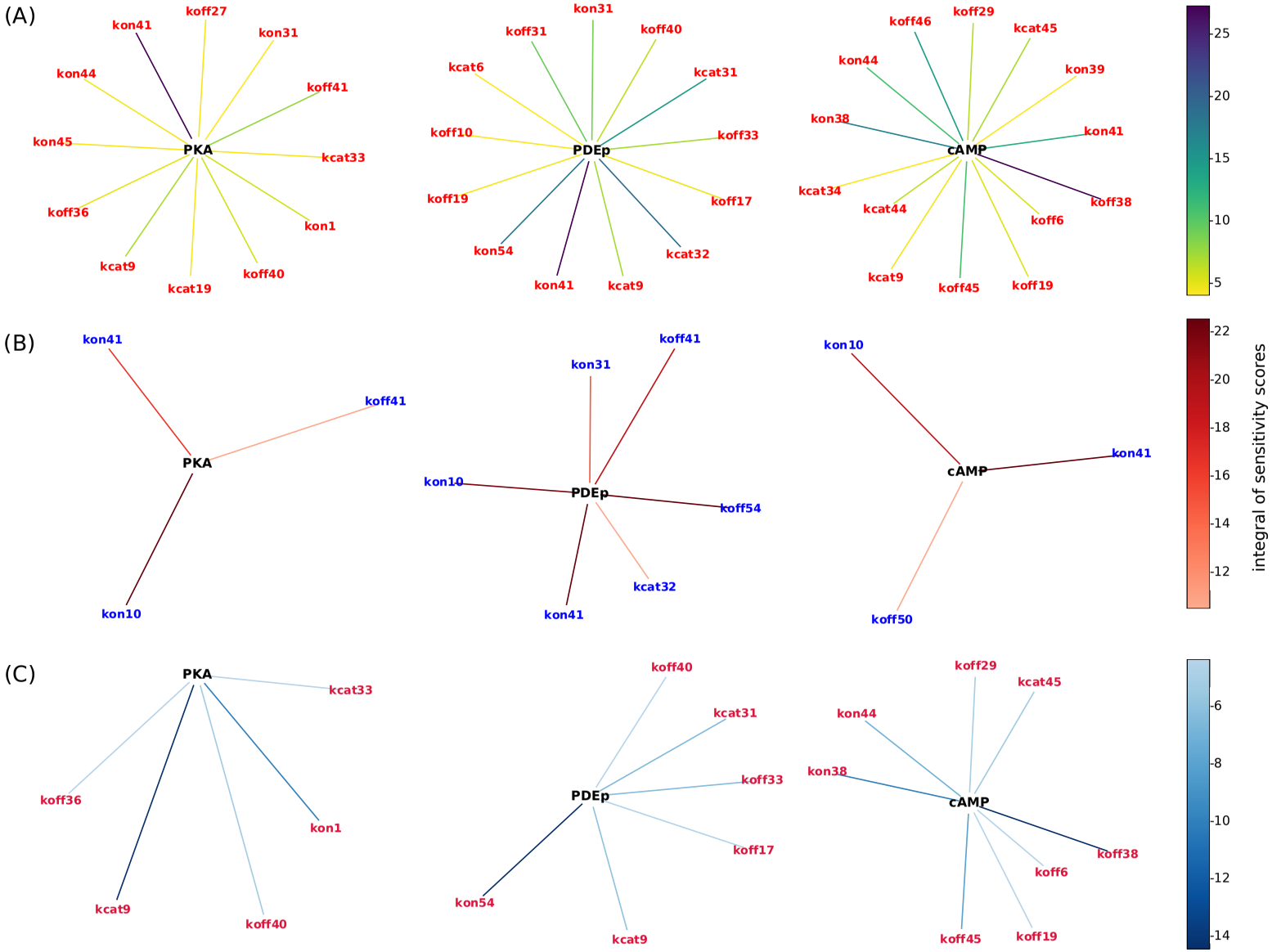


FIGURE 3.40: Neighbourhoods of 3 observables shared by two conditions that form a negative feedback loop: (A) the wild-type network, (B) the gained-importance network, (C) the lost-importance network.

observables in the constSer137 model (FIGURE 3.40B), as these networks contain parameters absent in the wild type network. These parameters become important due to alteration of the Ser137 function that affected the three observables. Among these parameters are "kon10", found in all three observables, "koff50" in the "cAMP" network, and "koff54", "koff41" in the PDEp network.

The first three parameters ("koff50", "kon10", "koff54") define rate of reactions with PP2A as one of the reactants. "koff50" is a constant rate of a dissociation reaction of a phosphorylated PP2A from DARPP-32, when PP2A is also bound to Ca^{2+}. "koff54" defines dissociation of Ca^{2+} from the PP2A, regardless its phosphorylation state. Earlier presented "kon10" determines a rate of binding between the phosphorylated PP2A and DARPP-32. Extended elaboration on potential reasons of this shift towards importance of PP2A in the constSer137, in particular in the phosphorylated form, was presented in the context of "kon10" in *Section 3.4.6.2*. The two other parameters ("koff54","koff50") confirm increase of importance of the phosphorylated PP2A for the observables of the negative feedback loop. This claim can be supported by analysis of parameters in the negative networks of "PKA" and "PDEp". Among parameters that lost importance for these two observables there is "kcat9" that parametrises dephosphorylation reaction of DARPP-32 at Thr75 by the unphosphorylated PP2A. The most distinctive lose of importance of this parameter occurred with respect to "PKA".

Shift in importance of these parameters demonstrate that, unlike the wild-type model, parameters involved in reactions of the phosphorylated PP2A gain greater control in the constSer137 model. This can also mean that the phosphorylated PP2A is more abundant than the unphosphorylated one, that is a result of increased availability of PKA due to nearly irreversible phosphorylation of Thr34.

Parameters determining reactions where PP2A is involved in, has already appeared in analysis of the top scored parameters of the wild-type model (*Section 3.4.4*). In particular, "kon54" was identified as important parameter for the "PDEp" observable. "kon54" is a rate constant defining speed of a binding reaction between Ca^{2+} and PP2A. According to the rule specification, when Ca^{2+} is bound to PP2A, PP2A has 4-times lower dissociation rate from one of its binding partners, DARPP-32. Therefore, a complex of PP2A and Ca^{2+} hinder binding of PKA to DARPP-32. What is more, as PP2A and PKA are binding

partners themselves, when PP2A has a free binding site, it can also compete with PDE to bind PKA. All together, when PKA is less likely to bind PP2A or DARPP-32, then availability to bind and phosphorylate PDE is enhanced. However, in the constSer137 model, the importance of "kon54" clearly dropped with respect to "PDEp" (FIGURE 3.40C) but a parameter of the reverse reaction, "koff54", became important for the same observable (FIGURE 3.40B).

Beyond the PP2A context, high difference in integral of sensitivity indices between parameters exposed other two reactions being in reverse to each other and parametrised by "koff41" and "kon41". These two are most crucial parameters for PKA in the wild-type network, that gained even greater control over this observable through the mutation of Ser137. "kon41" and "koff41" are constant rates of re-association and dissociation of PKA from R2C2, respectively. As inactive PKA is a heterotetramer that consists of two regulatory and two catalytic subunits denoted by R2C2, activation and deactivation of PKA takes place by dissociation and re-association reactions. As PKA is among reactants, these two reactions directly influence abundances of "PKA". Therefore, there is a clear and obvious relation between "PKA" and parameters of these reactions, "koff41" and "kon41". Other observable sensitive to changes in these reactions is "PDEp". However, only in the constSer137 model "PDEp" is sensitive to both parameters. The wild-type model, though with a high level, the observable is sensitive only to "kon41", that is even more elevated in the constSer137 model.

Taken together, two types of observable scoring systems were tested on the perturbed model. Because the number of identified clustering types was low, and to preserve consistency in applied measures between two model conditions, the observable score that included measures derived from all clusterings was used to prioritise observables of the perturbed model. Ordering of observables with respect to the score exposed shift in importance of observables between conditions that are consistent with the major effect of the perturbation. Among top scored observables, 3 were identified as important despite the perturbation. All together, 7 observables were prioritised for which integrated sensitivity indices were calculated with the HSIC-method. Different parameters were indicated as important with respect to different observables or their groups. Shift in controlling parameters between the two model conditions revealed growth of importance in the perturbed model of the phosphorylated

PP2A with respect to the observables forming a negative feedback loop regulating cAMP. Another indicative of the perturbation is activation of PP2B with respect to observables representing DARPP-32 phosphorylated at Ser137. Regardless the model condition, a parameter determining rate of PKA deactivation received the highest score with respect to observable representing PKA. To compare two model conditions with respect to obtained measures of importance, parameter and observable relations are summarised in a network structure for each condition. Parameter sensitivity scores are represented as edge weights joining observables and parameters. To find divergences between conditions with respect to multiple model outputs, edge weights of the perturbed model are subtracted from corresponding edge weights in the wild-type condition. Obtained in this way differential networks represent gain and lose of importance due to mutation of the Ser137 site. Examination of these networks revealed that for observables representing various forms of DARPP-32, half of parameters that were also highly scores in the wild-type model preserved importance in the perturbed model but for a set of observables representing different configurations of DARPP-32. For 3 observables that were commonly present in the two model conditions, analysis of differential networks has further emphasised the increase of importance of parameters defining reactions related to a phosphorylated PP2A. This result can be interpreted with mechanisms encoded in the model and the major effect of the constSer137 mutation.

3.5 Discussion

Practical exploitation of modelling studies in clinical application should allow to combine work of experimentalist and computational biologists in an iterative feedback loop. This could not be achieved without in-depth examination of control parameters and exploration of molecular mechanisms encoded in the model [232]. In particular, advent of new modelling techniques ought to be accompanied with novel methods of their analysis to guarantee use of their potential. RB modelling approach remains a niche in the domain of molecular modelling [129]. Extending ways of how RB models can be explored, what often defines strength of ODE-based modelling, could increase its application.

This motivation has led to propose an extended and automated analysis of RB models that results are presented in this chapter in form of a pipeline. The pipeline aims to partition and score observables, either defined by the modeller

or gathered from snapshots during the simulation. Selected observables are passed to GSA to identify groups of parameters that are important to them. RB modelling offers a new perspective in modelling of molecular systems. It was created to address the character of signalling systems, understood as parallel transient interactions between autonomous agents. Molecular species and complexes that are created during the simulation is one of particularly eminent novelties introduced by RB modelling. Not all molecular species are equally abundant and therefore, not all are important or informative. As seen, importance of particular species can be swung by perturbations applied to the model. Though with a great potential to accommodate detailed protein interaction data, results analysis and interpretation of such precise models might become problematic. In particular with growing scales of molecular models, what is an ultimate goal of automated model assembly. The question is then whether any groups of species can be prioritised as representative for the modelled system based on simulation results? Can we learn from the model what is worth to observe? These questions were a point of departure to use relatively recent approach of CorEx. Internal workings of CorEx relay on information theoretic objective that is optimised to learn a hierarchy of clusters that best explain dependencies between measured variables, i.e. observables. Results of clustering of single time courses showed that CorEx partitioned time courses of observables in justifiable way. Repeated CorEx executions with variable cluster numbers revealed that the major and strongest clusters have a stable set of members despite lack of convergence. Selection of members composing these strongest clusters can be interpreted as dominance of observables that respond to the sharp increase in the cAMP abundance. The impact of this cAMP pulse on copy numbers of proteins is much stronger compared to the Ca^{2+} spiking, as observed in selected time courses of the model (Figure 2.10). This interpretation was drawn from application of CorEx to time courses with two differently specified observable sets. These are hand-picked 19 observables and automatically collected 91 observables. The last ones being a total number of molecular species created by the model and established with recordings of molecular mixture during the simulation. As time courses of these observable sets are generated by the same model but differ in signal fragmentation, they should contain similar quantity of signal. As seen, CorEx identified equal proportion of signal in time courses for the two observable data sets as evidenced

by the prevailing score of normalised cluster strength allocated for the strongest clusters observed in the two observable data sets. As 91 observable definitions are exact compositions and configurations of molecular species that appeared during the simulation, the largest cluster of this data set reveals that only particular configurations of species are strongly dependent on the cAMP-signalling events or are present in equal abundances during the simulation.

In both observable sets, among observables of the strongest clusters are ones with closely matching expressions that could be represented with a single aggregated and generalised expression. Such observables were combined into one expression thereby reducing lists of original observables. This results with dimensionality reduction of the model output, another aspect of CorEx application often evoked by the authors of CorEx.

To incorporate measures produced during the CorEx evaluation of multiple time courses obtained with randomised sets of parameters, two types of observable scores were introduced. The first one is calculated with measures collected from all clustering. The second, with measures derived from identified clustering types and contains the clustering frequency term. As such, this observable score is reserved for results with distinctive proportion of clustering types found among all clusterings. No distinctive division into clustering types has been observed in time courses of randomised parameter sets, as there was only one recurring clustering type composed of 2 clusterings. This lack of agreement between clusterings could be due to the relatively hight variation of parameters (10-folds). To verify this interpretation, the same evaluation would have to be performed with time courses generated with lower parameter variability (5%, 10%, 30%). If this hypothesis is confirmed, then CorEx could be applied to test model robustness, alternative to what was presented in the context of GSA by Kent et al. [231].

Both observable scores indicated the same 7 observables as top scored ones that are directly involved in the cAMP signal. The same set of observables was indicated in the largest cluster when CorEx was applied to a single set of time courses of the model with the base-line parameter values. For this particular data set, large range of variability of parameters did not alter this particular clustering result. As the preliminary results yielded similarly partitioned observables, CorEx could be used for a single set of time courses to select a subset of observables, before the model is simulated with randomised

parameter sets. It would be especially advantageous for large RB models as the simulation time can be also computationally expensive when a large number of observables is tracked.

For the 7 identified observables, parameter sensitivity indices are calculated with HSIC. This method is a highly efficient importance measure of parameters that was introduced here as a model-free alternative to PRCC, proposed as sensitivity indices in the RB modelling framework by Sorokin et al. [272]. HSIC-based sensitivity indices are calculated for each time step of the simulation after the cAMP stimuli. The final sensitivity scores are obtained by taking definite integral of areas under sensitivity curves as a compact measure of parameter sensitivities. Though indirect, relations between highlighted groups of reactions with highly scored parameters are explicable with encoded mechanisms in the model. Analysis of sensitivity scores per observable revealed that less than a handful of parameters has significant effect on more than one observable. Moreover, less than 10 parameters, constituting a fraction of all 61 parameters, were found to have distinctive influence per a single observable. Addition of a negative control parameter revealed that the HSIC-based sensitivity indices are not free from artefactual sensitivity scores, evidenced by non-zero sensitivity scores gained by the dummy parameter. This aspect of the HSIC sensitivity indices has not been tested before, neither by the author of indices [282] nor in the study by Sinha [284], the first-time application of these indices in the context of molecular modelling. In consequence, as the HSIC sensitivity indices yield low levels of artefactual values, ones should not assume that every value of sensitivity above 0 reflects a real sensitivity. As it was suggested by Marino et al. [268] in the context of eFAST, sensitivity score of a dummy parameter could be used as a control group to statistically asses sensitivity of other parameters. By defining statistical significance of sensitivity scores in the pipeline, one could avoid using arbitrary cut-off for parameter selection in the network of integrated observables and parameters, and include only significantly scored parameters. One caveat is that this approach would require even more simulations of randomised parameter sets, a requirement that might become prohibitive with a large number of sampled parameter sets. Next to definition of statistical measure of significance for parameter scores, a similar need refers to observable scores. In practice however, with the large number of observables only the top ones can be considered in GSA.

Parameters and observables were combined into a compact network representation to facilitate analysis of parameter sensitivities with respect to multiple observables. As shown on the example of the 7 observables, one can have an immediate view on groups of parameters that are significant with respect to certain groups of observables, or parameter nodes and observables that connect otherwise separated groups, i.e. minimum cut nodes. Such commonly used measures and techniques defining relations in network graphs as centralities and clustering were not applied here as the size of analysed network was small enough to be visually examined. Network analysis is a dynamic research domain that studies relations in large data sets and therefore, it is a potentially notable advantage to apply graph-based techniques to analyse dynamical models. Weighted network representation was particularly chosen to define a method to investigate effects of perturbation induced in the based-line model. Comparison between different model conditions is an essential part of modelling-based studies. Moreover, application of the same measures composing the pipeline to two model conditions aimed to extend evaluation of consistency and significance of both techniques. Constitutive mutation of Ser137 was selected as an exemplary model perturbation. Ordering of observables with respect to the observables score calculated with measures collected from all clustering exposed shift in importance of observables towards molecular species of DARPP-32 phosphorylated at Ser137 that is consistent with the major effect of the perturbation. Among top scored observables, the ones composing a negative feedback loop were identified as persistently important despite the perturbation. Similar composition of the top scored observables was recorded for the observable score calculated with respect to a clustering type. Though the number of identified clustering types was higher than in the base-line model, counts of member clusterings in each type remained very low. Therefore, the other scoring method was used to prioritise observables to the next pipeline step.

Calculation of integrated sensitivity indices with the HSIC-method revealed growth of importance of a phosphorylated PP2A with respect to the observables forming the negative feedback loop regulating cAMP (“`kon10`”). For observables representing DARPP-32 phosphorylated at Ser137, a parameter indicative of the perturbation determines the rate of reaction activating PP2B (“`kon36`”). In both conditions, a parameter determining rate of PKA deacti-

vation received the highest score with respect to observable representing PKA ("kon41"). Growth of importance of the three parameters is consistent with size of impact of the cAMP pulse on abundances of other proteins, common in both model conditions, and domination of the DARPP-32 phosphorylated at Ser137.

These observations were derived from analysis of separate heatmap plots of sensitivity indices with respect to multiple observables, constructed for both model conditions. To study divergences between these conditions in a structured and unified way, differences in parameter sensitivities were defined as differential networks representing gain and lose of parameter importance due to mutation of the Ser137 site. Examination of these networks revealed that for observables representing molecular species of DARPP-32, different in each model condition, half of parameters preserved importance regardless the condition. Analysis of differential networks of observables shared by networks of both conditions has further emphasised the increase of importance of parameters defining reactions where the phosphorylated PP2A is a reactant. Another observation was that constant rates involved in activation of PKA gained even higher sensitivity scores in the perturbed model with respect to observable representing PKA and a phosphorylated PDE. This gain of greater control by parameters involved in the PKA activation caused by the constSer137 mutation can be interpreted by stronger dependence between these interactors due to increased availability of PKA to bind other partners.

Representing measures of relations between parameters and observables as network graphs demonstrated that they can be studied with different means than commonly applied clustered or ranked heatmaps. However, this particular method of analysis of differences between model conditions is an example relevant only to small networks and therefore, would not scale well for larger ones. More advanced methods for larger network comparison could be developed from methods already available in the domain of differential network analysis [294, 297]. Techniques of comparison between biological interaction networks defined for more than one condition, regarding different time-points, tissues, species or drug induced perturbations, have been commonly applied to protein-protein interactions and gene co-expression networks. These methods are mostly focused on measuring changes in the network topology caused by gain or loss of edges. This approach would not be

suitable in this study, as networks of observables and parameters are fully connected unless a measure of statistical significance of sensitivity indices would be provided to drop insignificant edges from the network. Nevertheless, there are also methods that are based mainly of weight statistics [297, 298] that could find a direct application in this study. For instance, a potential measure that could be used in further development of this pipeline is the Generalised Hamming Distance (dGHD) developed for weighted networks by Ruan et al. [299]. It has recently been improved by Mall et al. [300] and implemented as the "`DiffNet`" package in the `R` programming language. The proposed metric for measuring difference between two networks with the same number of nodes is enclosed within a normalised sum of squared differences of mean centred edge-weight between two clustering types. As this method was developed for the particular domain of genomic studies, one would have to examine if statistics applied in this method are also appropriate to capture the important regions of change in the observable-to-parameter networks. Therefore, though promising, application of this method in the context of this study requires to be evidenced by future research.

To improve the approach presented in this study, the difference network could be also applied not only to edges but also to node weighs defined by proposed observable scores. Calculation of the HSIC-based scores for all observables to obtain a complete network of weights between all observables and parameters might give a more complete view on alterations in modelled mechanisms caused by perturbations. However, the size of observable list defines the constraint of this approach.

Parameters of the RB model were equally varied over a range of ± 10 folds. Ideally however, prior knowledge about ranges of parameter uncertainty and their probabilistic distributions should be incorporated in the study design. For the model of Fernandez et al. [177], the domain of parameter variation could be defined within physiologically plausible bounds for the 35% of parameters that were based on experimentally characterised reactions by using error bars of replicated measurements. In case of fitted parameters, the spread of variation could be established by examination of the model behaviour under wide parameter ranges and setting the boundaries of parameter space that excludes these values under which the model fails to reproduce any characteristic features of behaviour. This approach would be particularly important

in parameter prioritisation to indicate most crucial parameters to be precisely measured in future. This would not be the case if the purpose of GSA was to measure the model robustness as this aims to establish the scale of perturbation necessary to abolish characteristic model behaviour. These aspects are worth to include in the procedure of GSA in the presented pipeline.

To improve the final part of the pipeline, where relations between critical parameters and observables are represented as networks, only indirectly linked rate constants could be preserved. These are constants that parameterise reactions where an observable is not present as a product or a reactant. This would allow for easier exploration of non-trivial relations.

Although the pipeline was particularly designed for the stochastic agent-based models, it is also appropriate for ODE-based model analysis as the particular method used for GSA is designed for deterministic models and used on averaged time series. Moreover, the RB model was refactored from the ODE model that represent an accurate approximation of these particular molecular behaviour.

3.6 Conclusions

In this chapter, I proposed a pipeline of automated exploration and analysis of RB models. Both presented measures of parameter and observable importance, HSIC and CorEx, provided sensible and justifiable results with respect to mechanisms encoded in two model conditions representing the wild-type and the perturbed phenotypes. Application of differential networks to compare two different model conditions offered an unified and compact representation of alterations in parameters and observables due to perturbation of the base-line model condition. Proposed method is limited to small networks and practical application would require more advanced methods for differential network analysis. Despite this limitation, this study provides a proof of concept to support analysis of complex dynamic models with graph-based techniques.

Having established that an automated pipeline for model analysis can offer legitimate results summarising different model conditions and phenotypes, now I would like to ask if model construction can be facilitated by recent advances in the domain of bioinformatics. As the RB language offers a molecule-centred perspective, it is aligned with bioinformatics data resources

that are concentrated on mostly protein- and gene-centred information. Therefore, in the next chapter, I examine what resources are available and relevant to RB modelling, and whether they could accelerate the process of building a dynamic model addressing any subject of biomedical inquiry in the current state of art.

Chapter 4

Exploring current resources for developing disease relevant rule-based models

4.1 Motivations

A common way of dynamic model construction requires extensive literature reading to assemble necessary details and evidence. The amount of effort involved in building models in such a manner is a limiting factor that restricts the model size and its subject of inquiry. Molecular models are composed of lists of biomolecular entities, reactions and parameters. From the bioinformatics perspective, these lists can be found in thematic databases cataloguing gene or protein-centred information that aggregate results from numerous experimental studies. Having at disposal a repository composed of reliable and relevant resources to dynamic modelling could accelerate the process of defining such models for potentially wider scope of biomedical inquiries such as identification of disease mechanisms. Therefore, identifying and assembling these resources beforehand could simplify and advance the process of modelling that target disease mechanisms. This assumption can be realistic in the light of recent advances in unification and standardisation of data formatting, access and annotation that are provided in machine-readable formats. To asses the validity of such assumption, first we need to identify such datasets and evaluate their coverage. This assessment is the main purpose of this chapter, performed with respect to an exemplary subject of biomedical inquiry. Having

a list of genes associated to a disease of interest, the question is how we would build a dynamic model by which mechanisms involved in this disease could be investigated?

As the relevance of data sources depends on the modelling framework of choice, this assessment will be performed with respect to the rule-based (RB) modelling framework. As elaborated in *Section 1.4*, the choice of this particular framework is not arbitrary as the RB modelling offers a modular, formal and concise method to capture protein interactions in a scalable way. Moreover, the protein and gene-centred focus of bioinformatics data resources stands in analogy to the molecule-centred perspective of the RB language.

In *Chapter 2*, based on an exemplary model, we learned what particular aspects of molecular mechanisms could be efficiently represented with a RB model. In *Chapter 3*, an approach to analysis of potentially large RB models was proposed. This chapter aims to clarify if up-to-date efforts in bioinformatics domain can facilitate such large model construction.

4.2 Introduction

The most important biomolecular entities in the cell signalling are proteins that are conducting the signal through interactions with other proteins, DNA, RNA and small molecules. Interactions between proteins has been a subject of intensive studies propelled by the development of a large range of experimental detection methods. This places the protein-protein interactions (PPIs) resources in the special interest of this study. PPIs provides evidence of existing associations between protein lists at different levels of directness. Identification of such associations is an important step as biological processes and functions are conveyed by interacting proteins. More detailed information regarding the actual mechanism of interactions requires a closer look at functional blocks of proteins, such as domains, motifs or repeats, and interactions between these protein subunits. As seen in *Chapter 2*, the RB framework is particularly designed for modelling complex interactions with site-specific details. Although, the RB language is flexible enough to define a model without explicitly stating what protein interfaces mediate the interaction, the site-specific information can clarify the exact mechanisms of reactions and impute hypothetical reactions.

Bringing site-level information and dependencies between protein in-

terfaces to the first plan allows to explicitly model impact of genomic variants, point mutations and deletions. Alteration of residues in motifs and domains, influence recognition and binding properties of these protein subunits [5]. This is particularly applicable in studies of disease-related mechanisms. For instance, Single Nucleotide Polymorphism (SNP) in the DNA sequence coding for a domain can lead to disturbance of interactions that the domain is involved when it is non-mutated [5]. This example locate the information about protein domain architecture and domain interactions in a special interest of this study.

One of the domain functions is to recognise and bind to amino acids with post-translational modifications (PTMs) [5]. PTMs are another important mechanisms employed in signalling systems. Among many variants, phosphorylation is the most abundant type of PTMs [230], with phosphoserine and phosphothreonine as the top two most frequently observed PTMs [301]. The ubiquity of this particular type of peptide modification could have been also observed in the limited example of the DARPP-32-network model that consists chiefly from reactions between phosphorylation sites and their enzymes. With respect to information availability, identification of phosphorylation sites and kinase-substrate relations belong to most frequently studied aspects of phosphorylation [16].

Having the model of DARPP-32, defined and explored in two previous chapters, the focus of this chapter could have been placed on reproduction of the system with use of variable primary resources. However, the DARPP-32 interaction network is an example of well studied system. The level of coverage of proteins in the pathway data resources, based on the example of REACTOME Pathway Database (REACTOME), can be expressed by the fact that out of 71785 proteins in the Human proteome[1], 10996 proteins can be found in any molecular pathway of REACTOME[2]. Therefore, concentrating on DARPP-32 network might not reflect the realistic availability of datasets for other biological mechanisms of interest. A more pertinent question would be if the assembly of relevant elementary data sets, related to any biological question, would bring us closer to the construction of a dynamic model. Therefore, with an attempt to recreate a common scenario of biological inquiry, this chapter is concentrated on a question regarding molecular mechanisms

[1] According to UniProtKB. Proteome identifier: UP000005640.
[2] According to the Version 63 released on December 18, 2017.

underpinning Attention Deficit Hyperactivity Disorder (ADHD), an example of highly heritable and complex disorder. The staring point of this inquiry is an assembled list of genes associated with the disorder. I present here a walk through the data sets that could be supportive in the RB model definition. First, the essential data sets are presented. Among these are: protein-protein interactions (PPIs), protein-domain interactions (PDIs), domain-domain interactions (DDIs), and kinase-substrate interactions (KSIs). Necessary mappings between resources and their coverage is examined from broader perspective of the Human proteome and the ADHD-associated genes that are assembled from three data resources of different provenance. To identify relevant functional modules in the ADHD-associated gene list, clustering and gene-set enrichment analysis is performed.

4.2.1 Protein interactions

Information of protein interactions can be derived from low and high throughput experimental methods that provide variable levels of details, specificity, sensitivity and types of interactions they report. The interaction types range from binary to co-complex, from transient to stable, from weak to strong. Among commonly used high-resolution methods are X-ray crystallography and nuclear magnetic resonance (NMR) [302]. Less detailed but more commonly used detection methods are yeast two-hybrid (Y2H) [303], co-immunoprecipitation of protein complexes (Co-IP), protein-complex affinity purification-mass spectrometry (AP-MS), tandem affinity purification (TAP), pull-down and protein chip technology [302]. Reliability of these methods varies significantly [304]. Comparative studies of the two most popular high-throughput experimental approaches, Y2H and Co-IP, report very small overlap between the two methods [305]. This poses general difficulties in comparison, integration and validation of reported interactions.

To ameliorate this situation, computational approaches for interaction prediction has been proposed independently, or as a support for experimental methods [306]. These computational interaction detection methods leverage different aspects of biological information, among which are protein sequences, structures, homology and protein-domain interactions [306, 307].

The PPI data resources can be divided into primary databases that collect and collate information directly from published PPIs detection experiments

(e.g. DIP, BIOGRID, IntAct), and meta-databases that integrate information from multiple primary databases and individual studies, e.g. Human Integrated Protein-Protein Interaction rEference (HIPPIE) [308].

Integration and functioning of such aggregated databases is largely dependent on compatibility and unification of data formats, annotation standards and curation systems. To address these multiple facets, the Human Proteomics Organisation (HUPO)'s Proteomics Standards Initiative (PSI), jointly with PPI data providers, proposed a community standard data model to facilitate the exchange, integration, analysis and verification of molecular interaction data between multiple resources [309]. This standard data model includes accepted data attributes and controlled vocabularies contained in the PSI-Molecular Interactions (MI) ontology. The ontology provides structured and unique identifiers to denote variable categories[3]. Among others, the ontology terms designate interaction detection methods (e.g. experimental interaction detection, X-ray crystallography) and interaction types (e.g. genetic interaction, phosphorylation reaction). Ontology terms are hierarchically ordered in parent-child and sibling relations to each other. A parent term designates more generic term than its children terms. For instance, the category of "interaction type" (id: `MI:0190`) is a parent term to "molecular association" (id: `MI:2232`), that is a parent term to even more specific "carboxylation reaction" (id: `MI:1139`). To guarantee data compatibility, the MI standard vocabulary is accompanied by guidelines for reporting experimental results, e.g. Minimum Information about a Molecular Interaction eXperiment (MIMIx). These all standards are enclosed in the PSI-MI database-independent data format. Among the PSI-MI standards compliant data providers are BIND, DIP, HIPPIE, BIOGRID, IntAct and MIPS. PSI-MI became a widely recognised standardisation initiative that has extended its facilities to encode other than PPIs forms of molecular interaction data, like nucleic acids, chemical entities, and molecular complexes [310]. The development of PSI-MI was followed by establishment of the International Molecular Exchange Consortium (IMEx), cooperation between large public protein interaction data providers to unify standards for data curation and collect non-redundant records of interactions in the PSI-MI format. To support computational accesses through web services and query languages to a number

[3]Molecular Interaction ontology website on Ontology Lookup Service (OLS): `https://www.ebi.ac.uk/ols/ontologies/mi`

of interaction repositories in the PSI-MI format, the HUPO-PSI introduced the Proteomics Standard Initiative Common QUery InterfaCe (PSICQUIC) service [311].

The PSI-MI interaction data format does not impose the use of common interactor identifiers. PPIs records are reported in multiple identifiers, not always overlapping between databases and referencing either to genes or proteins. A commonly acclaimed and used gene identifier across many data resources comes from Entrez Gene database of National Center for Biotechnology Information (NCBI). It is a database for gene-centred information providing unique, stable and tracked numeric identifiers for gene sequences and gene symbols [312]. As for the protein identifiers, the Universal Protein Resource Knowledgebase (UniProtKB) is a widely used resource of protein-centred information. UniProtKB consists of two sections: TrEMBL with automatically annotated proteins (unreviewed annotations) and Swiss-Prot with expert-curated annotations (reviewed annotations). These two types of identifiers for genes and proteins are used to cross-map PPI resources used in this study.

4.2.2 Protein domains and their interactions

The study of Schuster-Böckler and Bateman [7] found that interactions between domains are present more often in experimentally detected protein interactions than expected by chance. This finding suggests that some domain interactions are likely mediators of these interactions. Domains and their interactions have been used as indicators of potential interaction between proteins [7, 306, 313]. Identification of domains in proteins has led development of the protein classification system that divides proteins into families based on their domain architecture, providing means for protein characterisation. This classification system allows to functionally characterise newly sequenced proteins [3]. The domain identification and classification process is based on *protein signatures*. They are predictive models build on similarity between fragments of peptides that share local features (e.g. conservation at different positions) known to be associated with a function or structure [314]. There are multiple computational approaches that detect such patterns and define types of signatures, either based on the amino acid sequence or the 3D structure of a protein [3].

Major integrated resource of the protein-to-domain mapping is the Inter-

Pro Consortium database [315, 316]. It is a federation amalgamating multiple protein signature databases (CATH-Gene3D, CDD, HAMAP, PANTHER, Pfam, PIRSF, PRINTS, ProDom, PROSITE, SFLD, SMART, SUPERFAMILY, TIGRFAMs). InterPro provides an additional curation layer of member database records. Detected anomalies are reported back to federated databases [316]. These databases differ in methods and purposes of defining protein signatures. These differences obstruct mapping between signatures and complete integration of datasets. Methods relaying on the 3D structure, like CATH-Gene3D, are based on classification of known protein structures deposited in the Protein Data Bank (PDB) [316]. The sequence-based methods share the same baseline step of protein multiple sequence alignment that is used to build model of protein signatures. Searching for signatures is a major diagnostic method for analysis of novel sequences, protein families definitions as well as domain and functional site identification. Each protein in the database can be matched to multiple signatures. The examples of signature methods are position-specific scoring system (Hidden Markov Models, e.g. Pfam) [317], fingerprints, profiles (scoring matrix, e.g. PRODOM), and patterns (regular expression, e.g. PROSITE). In all member databases, the similarity in signature between proteins is used to define hierarchies of protein and domain families. On a monthly basis, InterPro analyses and annotates all protein sequences of UniProtKB with signatures of member databases with the InterProScan software package [318].

In this study, the major focus is on details of protein interactions and therefore, domain-domain interactions (DDIs) resources are of particular interest. Among multiple existing domain identifiers, the only one used in the DDI resources are accessions of the Pfam database. Therefore, this database is a main focus of this study. The Pfam database of protein families is continuously active since its establishment in 2003.

Th most reliable methods for the DDI identification are based on known 3D structures of protein complexes. This information is accessed from the PDB repository of 3D structures of proteins, nucleic acids, and complexes [319]. The DDI datasets derived in this way are collectively called gold standard domain-domain interaction (GSDDI) datasets [320]. Constituent datasets gold standard domain-domain interaction (GSDDI) are iPfam [321], 3did [322, 323] and PiNS [324]. The high-resolution structural information derived from 3D protein crystal structures does not provide significant coverage for existing proteomes.

Therefore, computational methods for domain predictions has also been in development [325]. What follows, there have been initiatives to integrate these two different data sources, with support of scoring systems helping to evaluate predicted data sets with respect to GSDDI. Among them are DOMINE [326], UniDomInt [327], DIMA 3.0 [328] and IDDI [320]. IDDI is the latest of four that aggregated data sets of its predecessors. The database integrated 23 datasets, including the three GSDDI and 20 computationally predicted DDI. IDDI was reused by other studies, e.g. for protein scaffold prediction [329] and in analysis of domain interactome of virus-host relation [330]. In this study, IDDI is used as a baseline of DDI information being the largest DDI dataset known to the author.

4.2.3 Phosphorylation sites and kinase-substrates

Although, data has been gathered on a range of PTM sites, most studies are focussed on protein phosphorylation (phosphoproteomics) [331]. It is the most ubiquitous and conserved type of PTM [14]. Rapid increase in amount of available data is mainly a result of advances in high-throughput methods based on mass spectrometry (MS), coupled with novel enrichment techniques. This has had a significant impact on the field of signal-transduction research [14, 332]. Experimental studies are also devoted to discover kinase-substrate relationships, mainly based on *in vivo* kinase assays and perturbation experiments [333]. The experimental techniques needed to obtain phosphorylation data are either expensive (based on MS [230]) or low-throughput (antybody-based western blots [334]). To address such experimental difficulties and low coverage of kinase-substrate relationships, there have been efforts to develop prediction algorithms for kinase-specific phosphorylation sites or phospho-binding motifs. These methods are chiefly based on pattern recognition [230]. For a review of computational techniques see Trost and Kusalik [335].

Though, the above picture suggests abundance of information on protein phosphorylation, there is incomplete representation and bias towards well studied proteins [331]. Furthermore, although a fairly complete picture of phosphorylation processes is composed of phosphorylated substrate, kinase and phosphatase enzymes that catalyse the reaction, and phospho-binding proteins that bind to the phosphorylated residue of the substrate protein, abundances of information on the two latter are rather scarce [16]. The progress and

research interest in the substrate-kinase networks is more evolved [336–338]. This study concentrates only on the substrate-kinase relation. This is particularly motivated by the fact that it can give us a view on proteins associated by concrete reactants, as opposed to phosphorylation sites alone. Moreover, from the perspective of dynamic modelling, a view on coverage of phosphorylation sites to proteins might be misleading as there are non-functional phosphorylation sites that do not have any regulatory role [16]. Identification of functional phosphosites was recognised as one of challenges of phosphoproteomics data sets [16].

Most of persistently updated repositories of PTMs are mainly concentrated on phosphorylation sites. However, alongside the PTM information, there are data sets of kinases-substrate relations. Among the largest and stably developed ones are PhosphoSitePlus [339], PHOSIDA [340] and Phospho.ELM [341]. To the author's knowledge, available sources are immature and scattered across multiple repositories. An effort would have to be put forward to integrate these datasets before working with them in a high-throughput manner. Therefore, this study concentrates on a single repository of PhosphoSitePlus® as a representative dataset. It is a database of manually curated resources of experimentally observed PTMs. The database mainly contains Human and Mouse proteins [339]. The release used in this study covers 53928 UniProtKB accessions[4]. Datasets are updated every six months.

4.2.4 Molecular pathways

Molecular pathways have been traditionally organised as diagrams of pathway maps composed with extensive curation and expert knowledge. To make possible programmatic analysis, other formats have been introduced such as Biological Pathway Exchange (BioPAX) [98] and Simple Interaction Format (SIF). The former encapsulates all details in a rich eXtensible Markup Language (XML) data structure whereas the latter, simplifies pathway information to represent it as graphs. Even simpler format than SIF is often employed in bioinformatics analyses, that is as gene sets, where each pathway is associated with a list of gene or protein identifiers [62].

Molecular pathway knowledge-bases are important resources for dynamic modelling as pathway maps contain the first layer of information re-

[4]Accessed 2018-01-24.

quired to construct a dynamic model. Pathway maps represent a topological network structure defined with detailed qualitative information about reactions that occur between molecular entities. The importance of information stored in pathway databases as a starting point of model construction can be reflected by the study of Büchel et al. [342] and Wrzodek et al. [343]. Büchel et al. [342] automatically generated kinetic, logical and constraint-based models from pathway representations stored in KEGG and MetaCyc [344] (Path2Model [342]). Kinetic models were only created for metabolic pathways, with the support of additional databases to identify kinetic reaction parameters, e.g. System for the Analysis of Biochemical Pathways - Reaction Kinetics (SABIO-RK) [345]. Signalling pathways were translated to qualitative logic models that does not contain mechanistic details [129]. Though, more information is available for metabolic pathways, the authors reported that kinetic data collected from SABIO-RK only exists for 12.2% of all Human metabolic reactions [342]. Similar attempt to translate signalling pathways from the KEGG-specific pathway format to the Systems Biology Markup Language (SBML) modelling format was performed by Wrzodek et al. [343] but without supplementation of kinetic parameters. In both cases, pathway databases provided a step forward towards automated generation of models.

Section 1.3.2 mentioned difficulties in unification of pathway databases caused by multiple factors. Even for extensively studied pathways, e.g. Wnt signalling pathway, agreement between databases in constituent pathway elements were found as very poor [103]. Reason for these observed variations between databases cannot be unequivocally designated, as lack of shared vocabularies or other such technical reasons and biological variation could be equally considered factors [103]. As no unified pathway database exists, this study relays on a single but important pathway reference dataset, REACTOME Pathway Database (REACTOME). It is an open source, publicly available and manually curated database [346]. The database provides annotations of molecular entities to terms representing biochemical reactions and pathways. Pathways in REACTOME are hierarchically structured terms. A pathway can be composed of other pathways and itself be a constituent of higher order and larger ones. On the lowest leaf-level are reactions or reaction-like events, among which are binding, complex formation, transport or polymerisation. Molecular entities included in a pathway can appear in other parallel path-

ways and are extensively cross-referenced with other databases, such as PDB, UniProtKB, Gene Ontology (GO) and gene expression in tissue samples [346]. Among biological processes included in REACTOME are signal transduction, metabolism, chemical synaptic transmission, gene transcription and disease affected pathways [346].

To other category of resources containing molecular pathways are databases dedicated to mathematical models, mentioned in the review of signalling pathway databases by Khatri et al. [55]. The growing number of models in the past years has put pressure on development and use of standardised modelling languages paired with the establishment of model repositories. Among these initiatives that gained general acceptance are ModelDB [347], BioModels [184], Database of Quantitative Cellular Signaling (DOQCS) [212] and the CellML repository [348]. Common functions to all these databases are storage, of models that contributes to availability to the wider research community. In particular interest of this study is the BioModels database. BioModels is a public repository of quantitative models of biochemical and cellular systems. The database is unrestricted by the biological subject and stores physiological and biochemical models [349]. The database has been continually released since 2005. Deposited models are divided into curated and non-curated. The former originate from peer-reviewed publications and successfully passed the curation procedure. The latter are derived from published studies that either failed to pass the manual validation aiming to reproduce results provided by the original publication, or were generated from automated procedures (Path2Models [342]). Models are provided in numerous model encoding formats, e.g. SBML and BioPAX. Moreover, curated models in BioModels are compliant to the Minimal Information Requested In the Annotation of biochemical Models (MIRIAM), introduced to standardise annotation and curation of computational models [349]. As the only such model database, constituents of models deposited in BioModels are annotated with unique identifiers from other resources, e.g. UniProtKB, GO and REACTOME. This is an important and unique feature of this database that bridges bioinformatics resources and dynamic modelling. As exampled by a study of 30 models of synaptic plasticity by Heil et al. [45] (*Appendix D*), modelled molecules are often referred with commonly used names that does not refer to concrete experimentally defined protein sequences with known reference identifiers. Moreover, they tend to

denote protein families or protein multimers. Identification of molecular entities in such models can be performed only in non-automated way that hinders their use and analysis.

Another advantageous aspect of the BioModels database is that it is a member of the European Bioinformatics Institute (EBI) Resource description framework (RDF) Platform [350]. The platform provides unified access to query federated resources with the W3C SPARQL language [351]. Thanks to this unification of resources, a pathway in REACTOME can be annotated with BioModels identifier if there is a model of the pathway (for details see: BioModels Linked Dataset, Wimalaratne et al. [351]).

4.3 Data sets

This section introduces the major data sets used in this study, alongside necessary resources and procedures to map identifiers across different data sets. Data sets used in this study are composed of the list of genes associated to a chosen biomedical inquiry, that is ADHD, protein-protein interactions (PPIs), protein-domain interactions (PDIs), domain-domain interactions (DDIs), and kinase-substrate interactions (KSIs). Cross-reference datasets are used to map genes to proteins, and proteins to domains. Genes are represented as symbols and Entrez Gene identifiers (Gene IDs).

Firstly, it is established if there are direct PPIs between members of the ADHD-associated gene list. The integrated data set of PPIs with only direct interactions are collected from 3 databases and assembled into a single set. Next, to learn more details about protein interactions found among the ADHD-associated genes, they are mapped to DDIs resources. In this step, a comparison between PPI and existing potential interactions resulting from domain-level information is performed. The DDIs are collected from the IDDI database, updated with resources that have ongoing releases. Because DDIs are encoded with a specific type of accessions derived from the Pfam database, the final results are exclusively based on the contents of this database. Another data set that the list of ADHD genes is examined with is kinase-substrate relations deposited in PhosphoSitePlus®. These dataset is no presented in this section as the dataset does not require processing and is used in its original form.

4.3.1 Disease gene set

ADHD is a neurodevelopmental disorder that mechanisms are investigated as an example of biomedical inquiry. ADHD is a chronic, complex and polygenic disorder. It has relatively high worldwide prevalence (children 5-8%, adults 3-5% [352]). Though, multiplicity of variable etiologies is involved in ADHD, including neurobiological and environmental factors, it is thought to be predominantly caused by biological factors (heritability estimation – 76%) [353], with an autosomal dominant mode of inheritance [354]. ADHD is characterised by clinical and etiopathogenic heterogeneity where underlying genetics has yet to be understood [355]. ADHD is composed of two main subtypes: inattention and hyperactivity-impulsivity. Although the only "biomarkers" of ADHD are behavioural tests, a range of animal models exists that mimic multiple deficits characteristic to ADHD. They are successfully attenuated by psychostimulant administration, similarly to affected Human individuals [356, 357]. Among rodent models of ADHD are knockout (DAT gene), tansgenic (SNAP-25 gene, TRbeta1 receptor) and inbred strains. Compared to other disease of CNS, in particular neurogenertive disorders like Altzheimer's and Parkinson's disease, ADHD does not have a clear molecular correlate that would easily indicate molecular-level mechanisms worth to be closely studied. As such, the choice of ADHD potentially avoids bias towards well studied diseases. In any case, ADHD is an exemplary disorder with known list of susceptible genes indicated by multiple studies (candidate gene, genome-wide common variants, genetic linkage and pharmacogenetics [358]) that effects could be explored in a larger pathway context and then mechanistically modelled, what mainly motivated this choice. Moreover, a popular theory about genetic etiology of ADHD is dopamine deficit in multiple brain regions that involve cortico-striatal circuits [359]. This is a particular link to the model of Fernandez et al. [177] that locates it as a potential component of a larger modelling project.

A list of candidate disease genes for ADHD is consolidated from three data resources. The first one is the ADHDgene database[5]. It is a high-quality manually curated community database assembled through extensive literature screening [86]. The list of genes is contained in the Core Data set, that is composed of full-text literature reading of ADHD studies retrieved from the

[5] ADHDgene website: http://adhd.psych.ac.cn. Accessed 2014-08-03 (not updated since then).

PubMed database, published between 1995 and 2014. Of the total of 364 studies included in the database, 16 are meta-analyses and 348 are marked as others[6]. Among the latter are genome-wide association studies, candidate-gene association studies, linkage studies, mutational studies, and genome-wide copy number variation analyses. The database enlists the total number of 359 genes, each assigned with 4 scores: a total number of studies, a number of statistically significant studies that confirm a gene-disease relationship as reported by a publication, a number of insignificant studies for this relationship and a number of trend studies that locates the significance values between these two threshold values. The significance value is dependent on the type of study. In the general candidate-gene association study and meta-analysis a significant gene have p-value < 0.05. In genome-wide association studies (GWAS), a significant association is assigned for p-value $< 1e-7$, non-significant for p-value $> 1e-5$ and trend for p-value between $1e-5$ and $1e-7$ [86].

To define the highest confidence set of ADHD candidate genes, the number of significant studies per gene is chosen as a quality criterion. It is dictated by examples of genes like CACNA1C that though intensively studied, are not confirmed by any statistically significant test result. With respect to the number of significant studies, out of 359 genes, 38 are confirmed in at least two significant studies (10% of the whole set) and 137 by at least one such study (38% of the whole set). These two gene subsets are used as high quality subsets and further referred as the *38-seed genes* and the *137-seed genes*.

The other two datasets used in this study are based on automated text-mining methods. The first one is MalaCards [360], a disease-centred meta-database that collects variety of information on more than 16 000 diseases, represented as "disease cards". MalaCards is a curated and automatically assembled platform gathering information from around 72 resources on different topics related to a disease. Among these topics are symptoms and phenotypes, drugs and therapeutics, publications and disease-associates gene sets. Disease gene sets for ADHD are derived from such resources as ClinVar [361], Online Mendelian Inheritance in Man (OMIM) [362], the University of Copenhagen DISEASES database [363]. ClinVar is a database of Human gene variations to medically important phenotypes [361]. OMIM is a database of high-quality and manually curated gene-to-disease associations [362]. The DISEASES database

[6] According to the search tab on the ADHDgene website.

contains disease-gene associations extracted from biomedical abstracts using text-mining techniques [363]. The total number of ADHD associated genes found in MalaCards is 83[7]. Each gene is associated with at least one of the three sources, OMIM (6 genes), DISEASE (73 genes) and ClinVar (4 genes). Provided gene scores were not taken into account.

The second dataset based on text-mining technologies is a gene-to-disease annotation dataset derived from the automated framework for creation of ontology-based annotations, OntoSuite-Miner [364]. The disease ontology used here is Disease Ontology (DO). The datasets of gene-disease associations are derived from publicly available datasets of gene-disease associations: OMIM [362], Gene Reference into Function (GeneRIF) [365] and Ensembl variation [115]. These three databases represent different approaches to disease-gene annotations. EnsemblVariation relies on genetic mutations like SNP, whereas OMIM and GeneRIF contain text annotations describing disease-gene associations. As mentioned earlier, OMIM is internally curated and high-quality database of relations between genes and phenotypes, whereas GeneRIF is a simple online tool that allows scientists to add the textual annotation of genes described in a publication. Annotation data in OntoSuite-Miner are processed using concept recognisers: MetaMap [366] and NCBO Annotator [89, 367] to identify terms found in ontologies, such as DO [368, 369]. To obtain a list of genes associated to ADHD, the OntoSuite-Miner annotation dataset is filtered with the DO identifier for ADHD (`DOID:1094`) with the result of 665 associated genes[8]. Majority of these retrieved genes is confirmed by Ensemble Variation (555). Much smaller fraction of genes is reported GeneRIF (115) and OMIM (7). There is a very low overlap between these three datasets: 9 genes are shared between Ensemble Variation and GeneRIF, 3 genes between GeneRIF and OMIM, and 1 gene between OMIM and Ensemble Variation. The union of the three resources is composed of 886 genes and constitutes the final list of ADHD-associated genes used in this study.

Figures 4.1 visualise an overlap between the three resources of ADHD-associated genes. Each diagram shows a different subset of the ADHDgene set, varied with respect to the number of significant studies that confirm association

[7]ADHD gene card website: https://www.malacards.org/card/attention_deficit_hyperactivity_disorder_2. Accessed 2018-01-18.

[8]OntoSuite-Miner github address: https://github.com/statbio/OntoSuite-Miner/tree/master/annotation_sources. Accessed 2017-02-11.

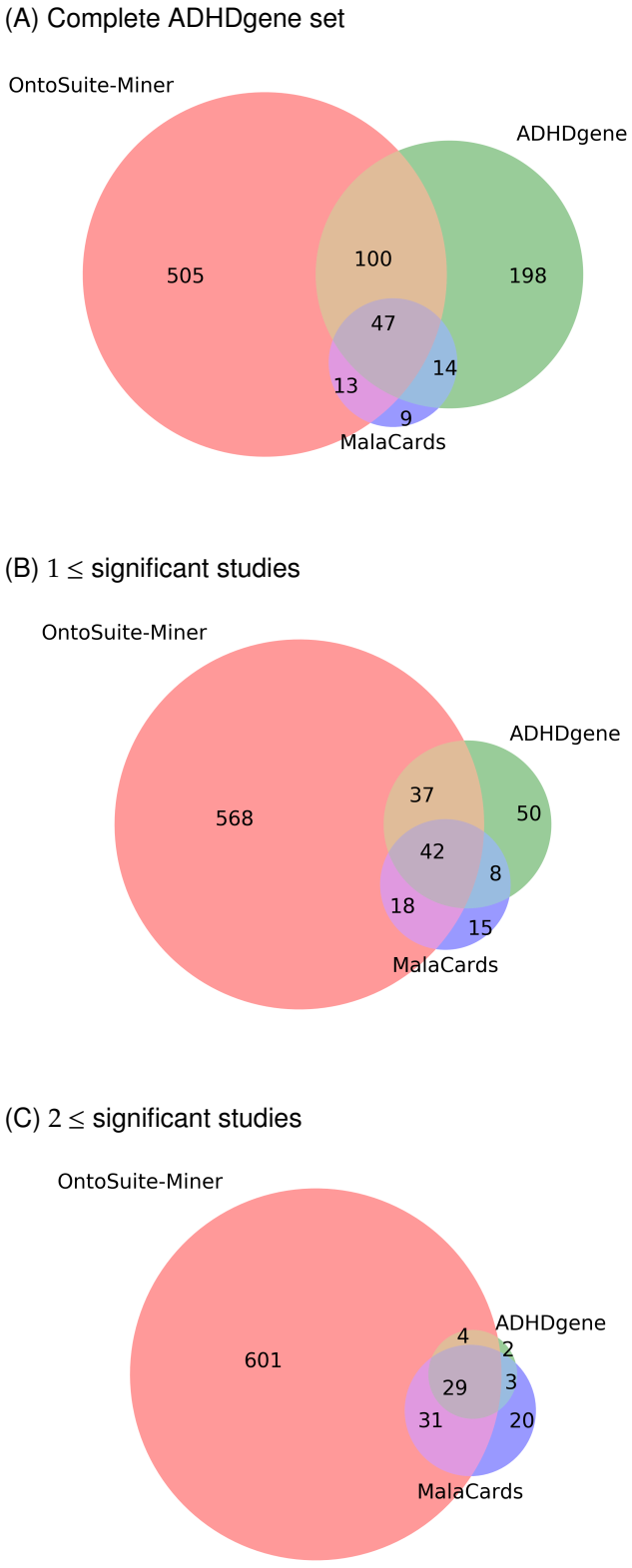


FIGURE 4.1: Representation of an overlap between three data sources for ADHD-associated gene lists with various subsets of the ADHDgene set, differentiated by the number of significant studies: (A) full set of 359 genes; (B) 137 genes confirmed by 1≤ significant studies; (C) 38 genes confirmed by 2 ≤ significant studies. MalaCards and OntoSuit-Miner gene lists are mainly based on text-mining methods, whereas ADHDgene relays on deep literature reading and curation.

of gene to the disorder. FIGURE 4.1A demonstrates that a lot of genes in the ADHDgene dataset cannot be found in either OntoSuite or MalaCards. This suggests that more than 50% of genes in the ADHDgene set are not commonly supported by other resources. When the list of genes in the ADHDgene set is constrained by gradually increasing requirement of the number of significant studies (FIGURES 4.1B&4.1C), the proportion of genes that are not found in either OntoSuite or MalaCards decreases reaching nearly complete coverage of manually curated genes by automatically assembled ones, for $2 \leq$ significant studies.

4.3.2 Protein interactions

Having assembled the list of genes connected by their association to ADHD, the next step is to ask if there are any direct interactions between them that would uncover more general biological contexts these genes are involved in. Therefore, the next essential step is to screen the list of genes with the PPI dataset.

The HUPO-PSI standardisation initiative for PPI datasets facilities the integration process of databases. There are three leading primary databases compliant with these standards: IntAct [370], BioGRID [371] and Database of Interacting Proteins (DIP) [372]. These databases provide data in two standard formats, PSI-MI XML and PSI-MITAB (tabular). Leveraging the convenience of the standard format for PPI datasets, PPIs are assembled from the three databases and unified within the PSI-MITAB format[9]. Although PSICQUIC provides a single-point access to all these resources, retrieval of interactions for more than a few hundred proteins appeared as inefficient and required flawless connection to the server and therefore, was not used in this study.

The PSI-MITAB format guarantees unified column contents and use of controlled vocabularies. Before consolidation of the datasets was possible, the database tables had to be parsed and cleaned individually with separate procedures. Of 15 obligatory columns, 7 were preserved. These columns report identifiers of an interactor pair, the PubMed database identifier of publication where the interaction was reported, taxonomies of interactors, an interaction detection method, and an interaction type. The last two identifiers are defined with unique MI-terms specified in the MI ontology. As interactor identifiers

[9]Datasets of individual databases were retrieved on 2018-02-07.

vary between databases, first they had to be converted to a common type of identifier to integrate the tables. DIP and IntAct databases use UniProtKB accessions (UniProtKB ACs) as their primary identifiers, e.g. P49418. On the other hand, BioGRID uses numeric Gene IDs, e.g. 1134. For the reason that UniProtKB protein sequences are used as reference for mapping domain information, UniProtKB AC was selected as a common identifier for the unified PPI dataset. To combine all three databases, BioGRID Gene IDs are mapped to UniProtKB AC. The mapping procedure was preformed with combined cross-references of Gene IDs and UniProtKB ACs, obtained directly from their primary providers, that is NCBI[10] and UniProtKB[11], respectively.

UniProtKB AC are divided into two sections, manually annotated Swiss-Prot and automatically annotated TrEMBL. As the former guarantees the high quality and non-redundant protein sequences, these are only preserved in the NCBI and UniProtKB final mapping files. The list of Swiss-Prot accessions was retrieved for the Human taxon from the UniProtKB database. It is composed of 20 258 unique UniProtKB ACs[12]. After removing TrEMBL accessions, the NCBI mapping file has 20 053 protein accessions mapped to 20 007 genes. The mapping file provided by UniProtKB has 18 935 proteins mapped to 19 125 genes. Union of both mapping files constitute the final bimap file composed of 20 063 proteins and 20 140 genes, with the total number of 20 482 gene-to-protein pairs. Unification of both mapping files revealed that there are 1461 gene-to-protein pairs that are present in only one of the two mapping sets. Among these, 1169 pairs are in the NCBI-derived file, and 292 pairs in the mapping file obtained from UniProtKB. This difference between cross-references identifiers confirms the insufficiency of using only the mapping data sets provided by one of the two databases.

Although DIP and IntAct databases use UniProtKB AC as primary identifiers for pairs of interactors and despite multiple identifiers supplied per interaction pair as aliases, not all have associated UniProtKB AC. These interactions had to be dropped in the process of table parsing.

[10]Gene ID to protein accessions mapping file: `ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/gene2accession.gz`. Accessed 2018-02-07. VALIDATED and REVIEWED positions were only selected as these are manually verified.

[11]UniProtKB Accessions to Gene ID mapping file: `ftp://ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase/idmapping/idmapping_selected.tab.gz`. Accessed 2018-02-07.

[12]Accessed 2018-02-08.

As the only one of the three, IntAct database provides information in an additional column if a binary interaction was generated by a spoke expansion of co-complex interactions. These interactions are retrieved as constituents of a protein complex [370]. The spoke expansion, next to the matrix expansion, is a method of converting n-ary interactions to a tabulated binary format. N-ary interactions are protein complexes that are identified in such experiments like tandem affinity purification (TAP). The spoke-model pairs a *bait* protein with each *prey* protein and therefore, a single complex is spoke-expanded into multiple interactions. As this procedure may generate false positives, interactions denoted as originating from spoke expansion were removed from IntAct records to obtain the most refined dataset.

The number of unique interaction counts for each dataset is 98085 for IntAct, 5508 for DIP and 281867 for BioGRID. A combined set of the three databases has 332190 unique interactor pairs between 31909 proteins. Figure 4.2 shows an overlap of interactions between the databases. The largest contribution is brought by BioGRID. This contribution could be lower if, similarly to IntAct, spoke-expanded interactions were identifiable and dropped from the dataset. Despite this large contribution, the overlap of BioGRID with other two databases is very low. The total percentage of non-overlapped interactions that comes from only one of the three databases is 84.63%. The intersection between the three databases is as narrow as 0.67%.

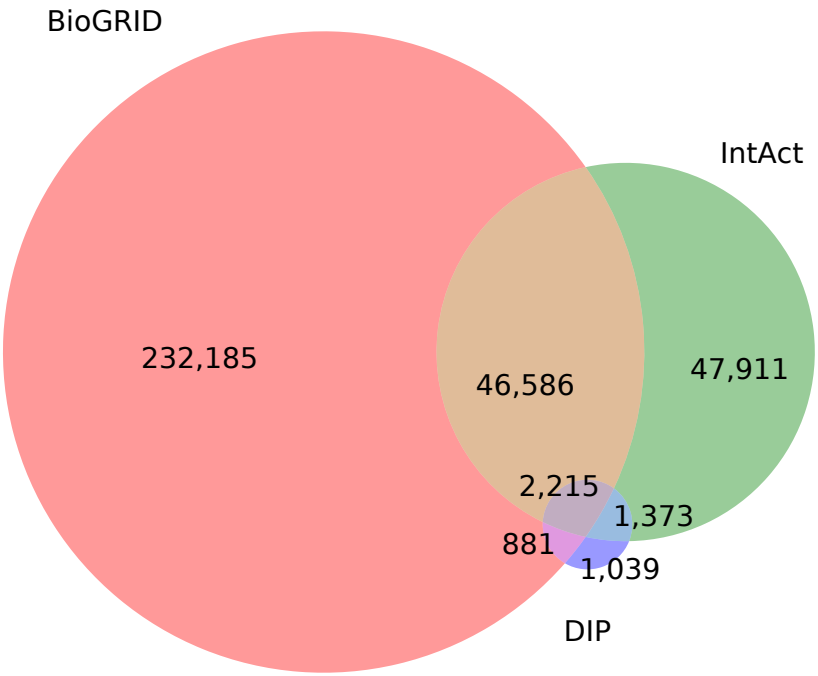
To obtain most refined data set of PPIs, only experimentally determined and direct interactions are included in the final dataset. The selection procedure was facilitated by the standard vocabulary of the MI ontology. It is worth noting that an interaction can be assigned with one or more interaction types and detection methods. This results in more records than actual unique interactions in PPI data sets.

The MI ontology provides two general terms to denote types of "direct interactions" (`MI:0407`) and types of "experimental interaction detection methods" (`MI:0045`). According the ontology, the definition of the former category is: "Interaction between molecules that are in direct contact with each other."[13]. The definition of the latter is: "Methods based on laboratory experiments to determine an interaction."[14]. Both categories have multiple child-terms that

[13]Definition of direct interaction term in the MI ontology: https://www.ebi.ac.uk/ols/ontologies/mi/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2Fobo%2FMI_0407

[14]Definition of experimental interaction detection term in MI ontologies: https:

(A)



(B)

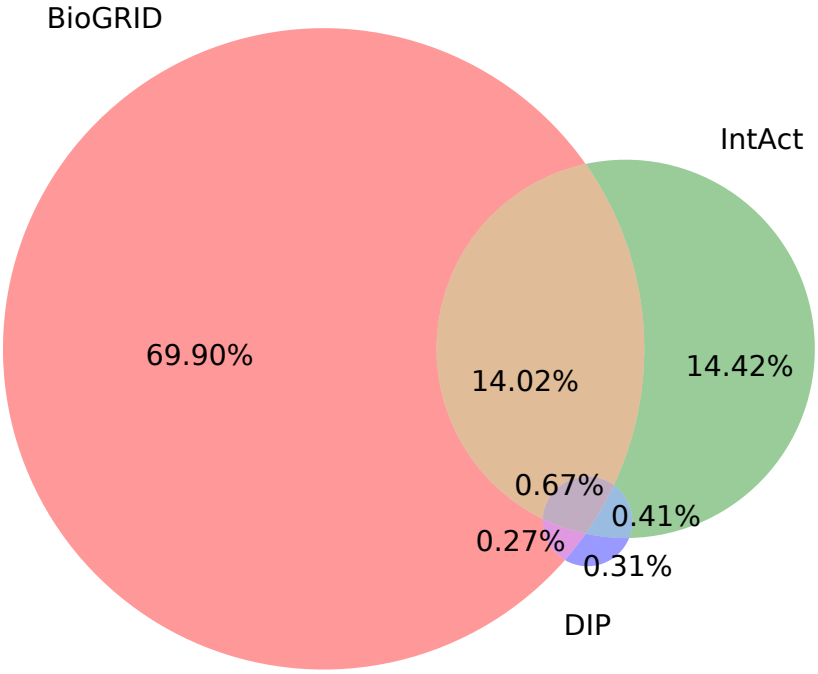


FIGURE 4.2: Overlap of interactions between three resources of PPIs: BioGRID, IntAct, DIP represented as: (A) counts; (B) percentages of the total number of unique interactions (332 190). The largest contributing dataset is BioGRID with the total of 281 867 unique interactions, then IntAct with 98 085 unique interactions and the smallest set belongs to DIP, 5508 unique interactions. The overlap between all three datasets is very narrow.

Database	Record counts				Interaction counts
			Human		
	Raw	Non-spoke	Genes	Proteins	
BioGRID	1 517 681	-	365 586	378 903	281 867
IntAct	794 921	500 881	-	186 225	98 085
DIP	288 612	-	-	19 665	5 508

TABLE 4.1: Representation of gradual reduction of dataset sizes before consolidation of the data sources. Columns are divided into record counts that represent rows in respective datasets and interaction counts. Starting from the left, the progression of columns shows the results of application of further modifications or filters on the datasets. Record counts imply duplicated interactions as one interaction can be associate with more than one identifier in other columns.

denote more specific concepts. The set of MI-terms representing each of the categories was obtained by parsing the MI ontology file[15]. Each set consists of the parent term and children terms. The total number of terms classified as "direct interations" is 69, whereas the total number of terms in the "experimental interaction detection methods" is 293. The terms were used to drop interactions that were annotated with terms different than in the set. The combined set of 332 190 interactions has 74 006 direct interactions, of which 73 115 interactions are experimentally confirmed.

As established by the example of IntAct database, interactions annotated with terms that belong to the set of "direct interactions", were also annotated as spoke expanded. Therefore, one could have limited trust to the remaining two databases, as they do not differentiate between spoke expanded and non-spoke expanded interactions. However, the databases were not further scrutinised as they represent leading resources in the field and contain the most current information available.

TABLE 4.1 demonstrates a gradual reduction of dataset sizes through the process of merging the three data sources. The first four numeric columns represent record counts, where an interaction between two the same proteins could be associated with more than one identifier denoting the publication,

//www.ebi.ac.uk/ols/ontologies/mi/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2Fobo%2FMI_0045

[15]The MI ontology file source: http://www.ebi.ac.uk/ols/index; version: 10-01-2018

the interaction type and the interaction detection method. Starting from the left, each next column is a result of applying further modifications or filters on the datasets. The first column presents the count of raw records before any filter was applied. The "Non-spoke" column represents the record counts after removal of interactions resulting from application of the spoke-expansion model. The "Genes" column represents the number of records of Human interactions represented with gene identifiers in the BioGRID database after converting the identifiers to UniProtKB ACs. The "Protein" column presents records after removing other than Human interactions and represented with the unifying type of protein identifiers. The last column represents unique counts of Human interactions represented with UniProtKB ACs. The consolidation of resources results in 578 443 records of 332 190 unique interactions.

4.3.3 Protein domain interactions

A major compendium of information about protein domains is InterPro. It joins 14 signature databases that apply different predictive models for identification of protein characteristics. Though DDIs are not represented with InterPro identifiers, reference to InterPro resources can serve as an overview of overall domain information.

Table 4.2 presents database-wise statistics reported on the 66.0 release of InterPro. The largest number of signatures belongs to the PANTHER database (90 742). However, the most integrated database with InterPro is the Pfam database (16 109). Table 4.3 shows counts of types of entries present in the release mapped to all UniProtKB sequences. Although the data set contains also information about other characteristics of proteins, e.g. binding sites and PTMs, the database is predominantly a source of information about protein families and domains. The percentage of reported counts of all UniProtKB proteins matching any signature in InterPro is 88.3%. Within UniProtKB data set, TrEMBL accessions are covered in 88.3% with any signature and in 80.9% with integrated signatures. Swiss-Prot is covered in 97.9% with any signature and 96.6% with integrated signatures.

The above statistics are reported across all species and types of UniProtKB accessions. The current release of Pfam (31.0), that provides unifying accessions for DDIs, covers 75.48% of sequences of reference proteomes in the UniProtKB database. Reference proteomes are composed both of reviewed and unre-

Signature Database	Version	Signatures	Integrated Signatures
CATH-Gene3D	4.1.0	2737	1366
CDD	3.16	12805	2535
HAMAP	2017_10	2216	2216
PANTHER	12.0	90742	7387
Pfam	31.0	16712	16109
PIRSF	3.02	3285	3223
PRINTS	42.0	2106	1974
ProDom	2006.1	1894	1307
PROSITE patterns	2017_09	1309	1289
PROSITE profiles	2017_09	1194	1161
SFLD	3	303	143
SMART	7.1	1312	1263
SUPERFAMILY	1.75	2019	1595
TIGRFAMs	15.0	4488	4445

TABLE 4.2: InterPro member database information for the 66.0 release. Source: InterPro release notes (23-11-2017)

Entry type	Counts
Active site	132
Binding site	76
Conserved site	687
Domain	8840
Family	20410
Homologous superfamily	2128
PTM	17
Repeat	278

TABLE 4.3: Types of entries in the InterPro database of the 66.0 release for all protein sequences in the UniProtKB database, with the total number of 32568 entries. Source: InterPro release notes (23-11-2017)

viewed UniProtKB accessions. Proteomes used as a basis for the last Pfam release originate from the 2016-10 release of UniProtKB. Number of matched proteins between InterPro resources and Pfam resources are not directly comparable as InterPro uses all UniProtKB accessions, whereas Pfam only reference proteomes. The number of all UniProtKB protein sequences is 161 521 for the Human taxon, whereas the Human reference proteome used to map Pfam accessions has 70 891 sequences. To asses the information gain obtained with InterPro, one would have to limit protein accessions used in mapping InterPro signatures to the same version of the Human proteome as used in the Pfam dataset.

Similarly to PPIs, proteins expressed in Human are the main focus of this study. Therefore, the Human reference proteome annotated with Pfam accessions was retrieved from the Pfam database[16]. The total number of 6116 unique Pfam accessions are mapped to 49 717 proteins of Human reference proteome. The number of protein in the proteome is 70 891 composed of Swiss-Prot (˜30%) and TrEMBL (˜70%) accessions. Of the 49 717 protein accessions mapped to at least one Pfam accession, 37% are reviewed (18 473)[17]. The number of Pfam accessions found in these protein subset is 6084. Similarly to InterPro, not only protein domains are in the database, though they comprise the majority of entries (Figure 4.3).

The Integrated Domain-Domain Interactions (IDDI) dataset is used as the source of DDIs. The dataset has not been updated since 2011. However, two included sources that belong to GSDDI datasets, are regularly updated, 3did[18] and iPfam[19]. These two resources were updated with preservation of the IDDI data format. The number of interactions in each GSDDI after the update is: 2898 (PINS), 11 200 (3did) and 9561 (iPfam). The reliability scoring system proposed in IDDI for reported interactions was not updated here and excluded from the dataset.

After adding the two updated GSDDI sources to IDDI, the number of non-redundant DDIs is 209 941, with 9126 unique Pfam accessions. Before the update, the IDDI dataset had 204 705 DDIs. Therefore, the update increased

[16]The file with the Human proteome annotated with Pfam accessions (release 31.0): `ftp://ftp.ebi.ac.uk/pub/databases/Pfam/current_release/proteomes/9606.tsv.gz`

[17]The list of reviewed UniProtKB accession was retrieved from the UniProtKB database on 08-02-2018. The same list was used for filtering the PPI dataset

[18]The 3did dataset File name: 3did_flat.gz. Version: 2017_06.

[19]Accessed 2014-07-22.

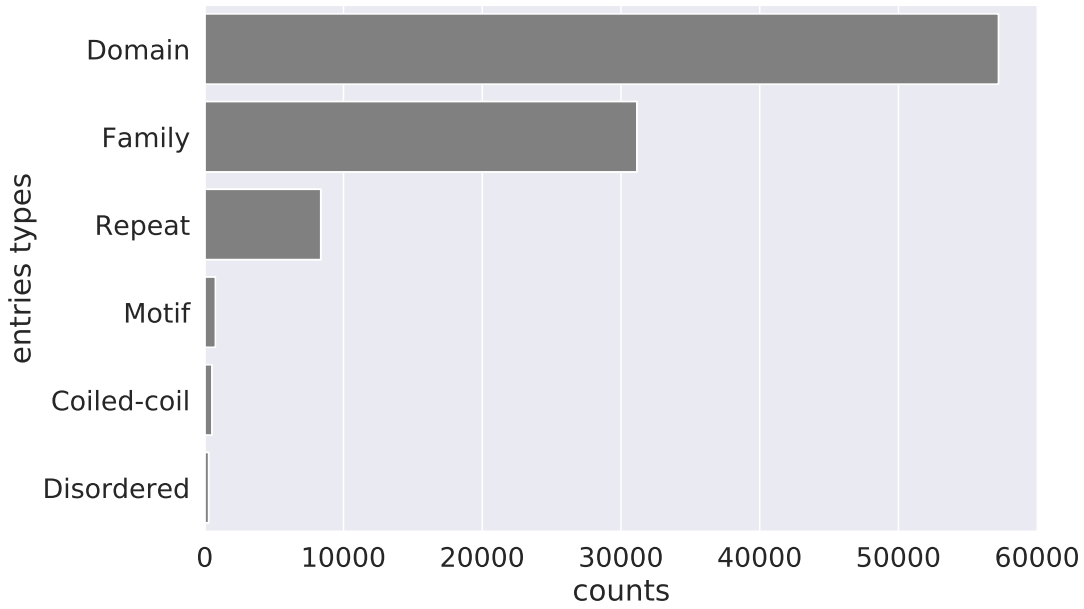


FIGURE 4.3: Pfam entry types that were found in the Human reference proteome. "Disordered" and "Coiled-coil" are most recently added types [373].

the number of interactions by 5236. The original IDDI version had 7351 Pfam domain accessions from the Pfam version 24.0, where the total number of signatures was 11912 (current 31.0 version: 16712). Pfam accessions in the updated IDDI dataset was filtered with 6116 Pfam accessions found in the Human reference proteome. These reduced the final non-redundant list of interactions to 130843, with 4271 unique Pfam accessions. The drop in the number of interactions and Pfam accessions was also caused by withdrawal of Pfam accessions by the database or by the fact that they were not found in any protein of the Human reference proteome. There was additional slight decrease of IDDIs by 47 interactions, as they were no longer supported by the updated data sets. The total number of interactions that are reported by at least one GSDDI is 7461 (5%). The number of interactions in each GSDDI data set is: 1650 in PINS, 6037 in 3did, and 5297 in iPfam. The number of domain interactions per source comprising the final IDDI is shown in FIGURE 4.4. FIGURE 4.5A demonstrates the number of sources per an interaction. Great majority of these interactions is confirmed by a single source. FIGURE 4.5B shows the same aspect of DDIs but for the subset of IDDI limited to the GSDDI resources.

The IDDI database did not differentiate between intra- and inter-chain

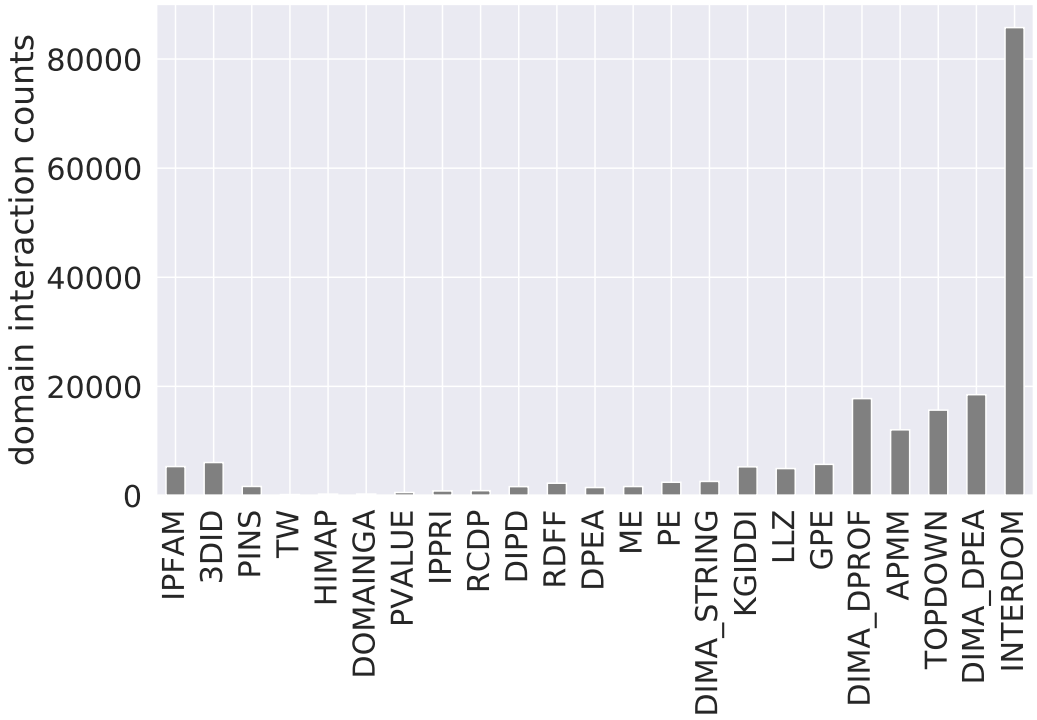
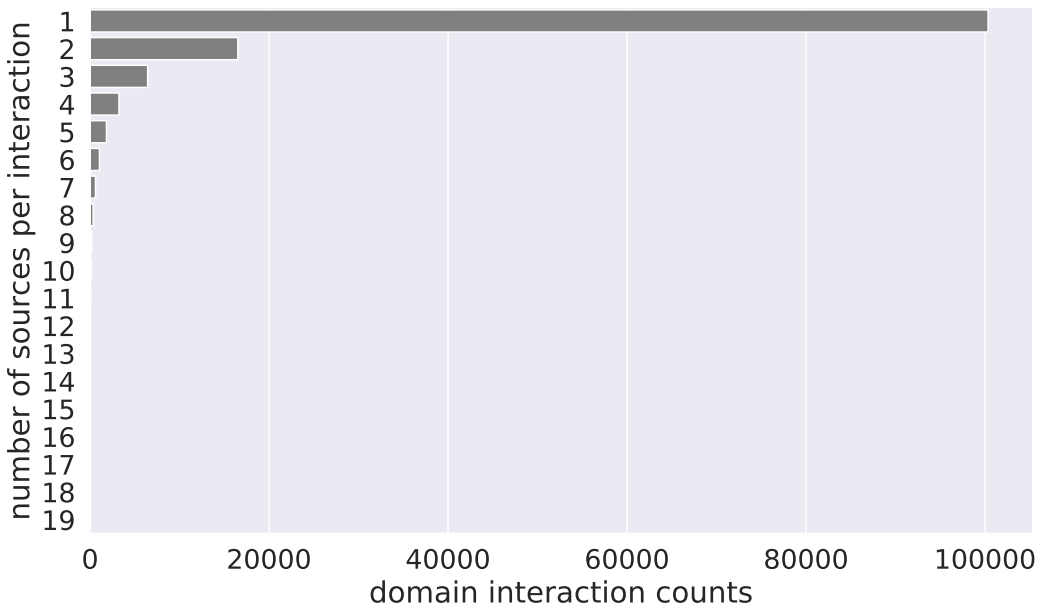


FIGURE 4.4: Number of protein domain interactions per resource that compose the IDDI dataset of DDIs. Golden standard resources are 3did, iPfam and PINS. Compared to computationally inferred datasets, e.g. INTERDOM, overall number of GSDDIs is very low.

(A)



(B)

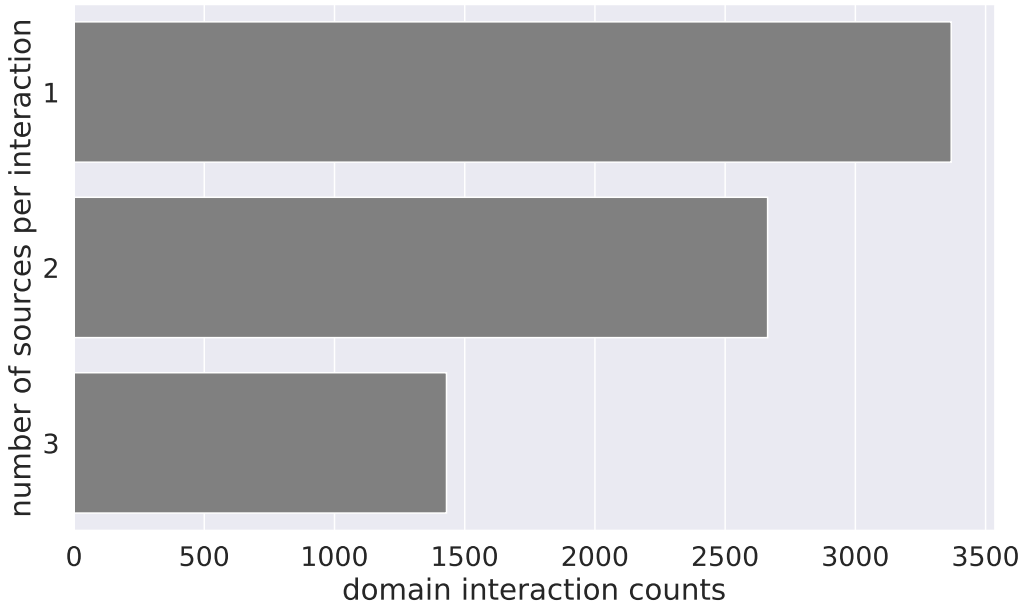


FIGURE 4.5: Distribution of sources per protein domain interaction, for (A) all 23 sources; (B) GSDDI sources: 3did (6037), PINS (1650) and iPfam (5297).

domain interactions. Though this distinction exists in two GSDDI databases (3did and iPfam), it is not taken into account in this study. This differentiation between intra- and inter-chain domain interactions is limited to existing examples of the PDB-derived crystal structures of molecular complexes. Following the example of the study by Schuster-Böckler and Bateman [7], it is assumed that the modular assembly of domains in evolution of proteins, that happens for instance through gene fusion/fission, can swap domain interactions between two categories. Therefore, first the accent is posed more on potentially existing interactions, leaving their refinement to later steps.

4.4 Methodology

Presented datasets were chosen to indicate functional links between the ADHD-associated genes. This section outlines steps undertaken to combine these datasets together with intermediate procedures to map between identifiers. Identified associations are represented with network graphs that are clustered to partition gene or protein nodes into strongly related subgroups. This section presents particularities of the selected clustering method. Enrichment analysis is performed with respect to library of pathway categories to determine if subsets of the ADHD-associated genes or proteins are overrepresented in any of these categories. Among many techniques of enrichment analysis, this study employs a special type of over-representation analysis (ORA) that take into account hierarchical relations between pathways.

4.4.1 Outline of steps

Figure 4.6 presents a diagrammatic representation of steps undertaken in this chapter. Gene IDs of the assembled list of genes associated to ADHD are mapped to UniProtKB ACs. These ADHD-associated proteins are used to filter the in-house PPIs to identify direct and experimentally confirmed protein interactions where both interactors are in the set of ADHD-associated proteins. Identified pairs of interactors are represented as a graph network that is subjected to clustering and clique identification. Aside from PPIs, the set of ADHD-associated proteins is also mapped to three other datasets. These are kinase-substrate interactions (KSIs), BioModels and proteins mapped to domains (PD). The latter dataset is combined with DDIs to identify interaction details on the level of protein subunits (Figure 4.7). These integrated datasets

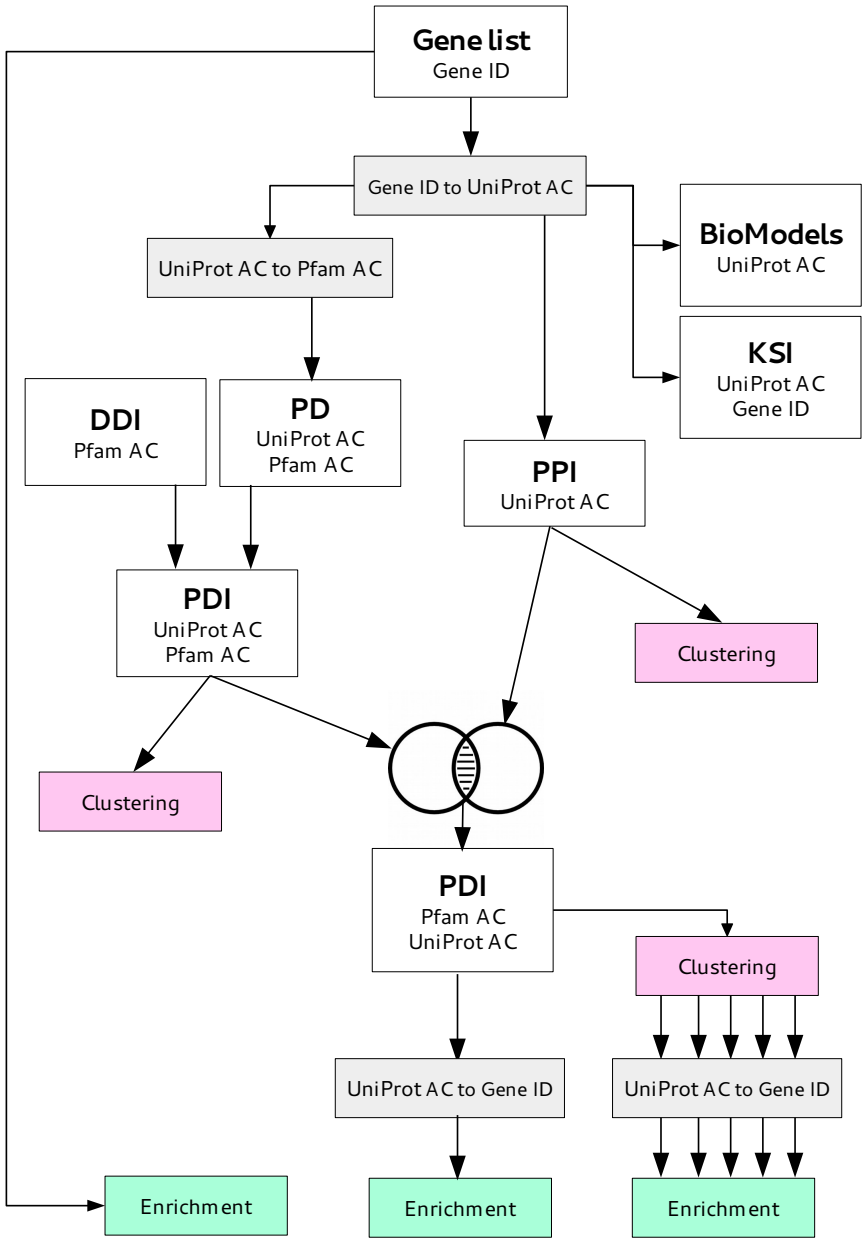


FIGURE 4.6: Outline of steps undertaken in *Chapter 4*, that include mapping the ADHD-associated genes to various resources with biological relations (white rectangles), intermediate steps of cross-referencing identifiers of different datasets (grey rectangles), and used techniques on these datasets (pink and green rectangles). The intersection icon denotes integration of two datasets, where a subset of shared entries is progressed to further analyses with clustering and enrichment techniques. White rectangles, representing datasets, contain information on types of identifiers used in the respective datasets. Abbreviations: DDI – domain-domain interactions; PD – proteins mapped to domains; PDI – protein-domain interactions; PPI – protein-protein interactions; BioModels – database of models; KSI – kinase-substrate interactions.

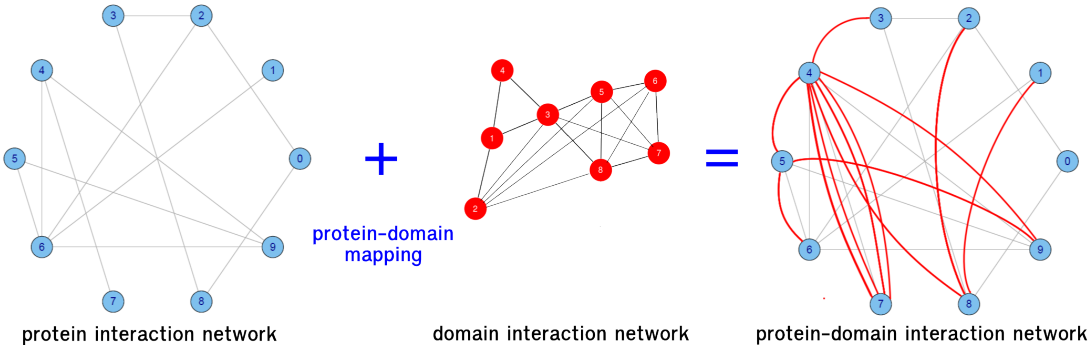


FIGURE 4.7: Schematics representing of enriching PPIs with DDIs. This procedure is applied to examine the number of potential domain-based interactions that could exist between proteins.

are represented as a network graph that is subjected to a clustering procedure. Intersection between PDI and PPI networks is taken to preserve these protein interactions that are mediated by at least one pair of interacting domains. This strictly limited subnetwork is subjected to clustering. In the last step, the original list of genes, associated with ADHD, is subjected to enrichment analysis with respect to gene-to-pathway annotations. The same procedure is performed on protein nodes of the stringent PDI network. Results of enrichment into pathway categories are reported for the whole set of network nodes and for each subset of these nodes, obtained with the network clustering.

Alongside presentation of coverage of the datasets with respect to the ADHD-associated genes, the coverage with respect to the Human proteome is also reported for selected datasets.

4.4.2 Network analysis

Network analysis allows to examine relations between numerous entities represented as abstract nodes or vertices. Differentially expressed or disease-associated genes are mapped to PPI resources to identify binary interactions proving close associations between identified molecules. PPIs are commonly represented as networks, with vertices denoting proteins and edges associations between them. Identification of densely connected components

or clusters among these molecules can point to important modules, involved in a common pathway or biological function. There are diverse approaches to cluster identification that often produce different clustering results for the same network. Computational complexity and running time are crucial aspects that are taken into account in method evaluation and selection. Scalability of clustering algorithm is particularly important in large-scale network clustering. Variable network structures, sizes and evaluation metrics prevent from categorical decision which clustering method is superior to others [79, 81]. Attempts have been performed to provide guidelines which clustering algorithm is most efficient and accurate in most circumstances [79, 81]. In particular, the study of Yang et al. [79] compared 8 state-of-art algorithms implemented in "`igraph`" package of `R` language with respect to accuracy and efficiency as most important aspects of successful algorithm. Comparison was performed on a benchmark generative network model of Lancichinetti-Fortunato-Radicchi (LFR) with a wide range of network sizes and values defining the *mixing parameter* μ of the network. This parameter, equally defined for each network node, is formulated as a ratio between the number of edges that connect a node to other clusters, to the total number of its adjacent edges [79]. The larger the number of edges shared with other communities, the task of community identification becomes more difficult as the community structure becomes obscured [79, 81]. Additional aspect found to influence the reliability of algorithms is the network size. In larger networks it is more difficult to correctly identify communities [79, 81]. Yang et al. [79] investigated dependencies between accuracy and computing time with increase of the mixing parameter and the network size. In general, all tested algorithms performed well with $\mu \leq 0.5$ and with the number of nodes ≤ 1000. In networks with more demanding community structures ($0.5 \geq \mu 0.6 \leq 0.6$) and the higher number of nodes but ≤ 6000, the "multilevel" algorithm outperformed other tested algorithms. This was found despite that algorithms based on modularity optimisation suffer from resolution limits [374]. As in this study variable network structures are clustered with different levels of separation between communities, the more flexible the algorithm is with respect to this property, the more favourable it is. The "multilevel" algorithm was proposed by Blondel et al. [82], and it is more commonly known as Louvain [81, 82]. The algorithm is a scalable method of modularity maximisation, that was originally prosed by Newman [80]. The

Louvain algorithm is a hierarchical approach with agglomerative strategy using greedy optimisation of modularity score [78, 82]. Initially, each network node is considered as a singleton community. The number of communities is gradually decreased as neighbouring nodes are aggregated. Aggregation of nodes is only possible if the modularity of the whole graph is increased as a result of such aggregation. When further increase in modularity score cannot be achieved, in the next phase the algorithm contracts identified communities into single nodes and repeats the same procedure on the reduced graph until modularity score cannot be further gained [82].

The Louvain method was also mentioned in the review of Javed et al. [78] as the one that outperformed other greedy algorithms for modularity maximisation. Emmons et al. [81] compared Louvain to other 3 algorithms with respect to different cluster quality metrics, and found that Louvain generally outperformed other popularly applied InfoMap.

In this study, this algorithm was favoured among others as it does not require the network to be fully connected, and scales linearly with number of nodes [82]. The Python implementation of the algorithm in the "`igraph`" package as the "`community_multilevel`" method was used in this study. Arguments of the method were left to default, meaning that the community structure that has the best modularity score is returned.

4.4.3 Enrichment analysis

A commonly used method to identify associations between lists of genes is an enrichment analysis, briefly presented in *Section 1.3.2*. Enrichment analysis informs if a set of genes of interest contains a significantly high number of genes associated to the same annotation in chosen category. This approach allows to gain a better understanding of underlying common biological processes found in the list of genes. Reference datasets supplying annotations are commonly organised as ontologies with hierarchical organisation of terms as direct acyclic graphs. The hierarchical organisation implies certain relations between terms, where the most generic terms are located at the top, and the most specific ones at the bottom. Moreover, the higher the term is located in the hierarchy, the more genes are associated to it as the parent term aggregates all genes from its child terms. Hence, a single gene can be found on different levels of annotation specificity. Depending on the purpose of the analysis, it

is important to be able to choose the level of retrieved terms. In the dynamic modelling context, the most preferable level is the most specific one, as it allows to retrieve relatively small and cohesive group of genes.

This problem has been addressed by the algorithm proposed by Alexa et al. [375] for GO, that allows to identify the most specific terms among significantly enriched ones. Alexa et al. [375] improved scoring of GO terms by including the underlying GO graph topology in term scoring. This approach removes strong correlations between neighbouring terms, a common feature of standard methods for the GO enrichment analysis.

Assuming that a child term is potentially more interesting than its more generic ancestors, significance of a term is calculated depending on the significance of its child terms [375]. In this way the enrichment of a more generic term is ignored, and less frequent low-level ones that are more specific and potentially more interesting to surface. The algorithm leads to more refined results than a set-based enrichment analysis that ignores the ontology structure. The algorithm is implemented as the "`topGO`" package in the R language but only for the GO reference database. For the purpose of identification of important disease mechanisms that could be dynamically modelled, pathway reference datasets are more informative as intensively studied and verified pieces of molecular mechanisms. They are also a usual commencing point in the process of construction of molecular models. Therefore, it is of interest to apply the same approach to the pathway-related gene annotation sets. This can be achieved with the "`topOnto`" package written in the R language [376] that is based on the "`topGO`" tool developed by Alexa et al. [375]. The package extends the advantage of the Alexa et al. [375]'s method to any hierarchically structured dataset, such as REACTOME. Pathways in REACTOME can be composed of other pathways. On the lowest leaf-level are reactions or reaction-like events, such as binding, complex formation, transport or polymerisation.

These hierarchically organised terms are transformed into a simplified ontology standard format, the OBO file. "`topONTO`" implements various test statistics and methods to eliminate local similarities and dependencies between ontology terms that are implemented in the "`topGO`" package by the follow-up studies [377]. In this study, the "`elim`" algorithm by Alexa et al. [375] is used, paired with Fisher's exact test as a measure of significance. The decision to select the "`elim`" algorithm was based on clarity of the number of comparisons

performed by the algorithm. This number was further used to correct for the false discovery rate with the Benjamini and Yekutieli multiple testing correction [378]. In the "`elim`" approach, enrichment analysis starts at the bottom of the ontology graph. If a child term is significantly enriched amongst the genes of interest, this influences the number of genes annotated to its ancestor terms. All genes associated to the enriched child term are removed from the ancestor terms leaving most specific ones with the minimal indicated significance.

A background dataset (N) is composed of genes of the Human genome. each associated with a set of REACTOME terms. The REACTOME terms were retrieved from the REACTOME database and filtered for the Human's Gene IDs[20]. The total number of terms is 1845, annotating 10045 genes.

4.5 Results

4.5.1 Protein interaction network

PPIs between the ADHD associated genes are assembled from the direct and experimentally identified dataset of PPIs (Section 4.3.2). To match the identifier type of the PPI dataset, the ADHD Gene IDs are translated to UniProtKB ACs. This mapping between identifiers is performed with the same dataset that was used to unify BioGRID with the other two PPI databases (Section 4.3.2).

Of the total number of 886 genes associated to ADHD, 746 Gene IDs are converted to 760 protein UniProtKB ACs. There are 140 genes that are not mapped to proteins. Almost all these genes (134) are non-coding DNA sequences that are not transcribed into proteins, with an exception of 6 that have unknown or withdrawn Gene IDs. Great majority of non-coding genes are non-coding RNA (ncRNA) (127), that are transcribed into functional RNA molecules. They can potentially have a regulatory function in various aspects of gene expression [379]. Among them, are 2 short nuclear RNAs (snoRNAs) that might play role in modification of other RNA forms [379]. There are also 7 pseudo-genes among the ADHD-associated genes. These are DNA fragments similar to existing protein-coding genes but with a modified open reading frame (ORF) that prevent their translation to proteins [380]. There are various types of pseudo-genes, some known to have a functional role in gene expression

[20]Entrez Gene ID to REACTOME term annotation file source: `http://reactome.org/download/current/NCBI2Reactome_All_Levels.txt`. Accessed 2016-11-19.

[379]. These pool of identifiers could be included in analysis; however, research on ncRNA is rather scarce and the information regarding the role of ncRNA might not be detailed enough to include it in the dynamic model.

Filtering of the integrated PPI dataset with the ADHD protein list resulted with 721 records representing individual pairs of potentially duplicated protein interactions. An interaction can be duplicated in this set if it has more than one publication identifier, type or detection method. The number of unique interactions is much lower, 371. They are composed of 270 proteins, of which 101 are self-interactions.

As all constituent PPIs resources report the type of interaction encoded with MI terms, the extend of information content on interaction details is examined in the ADHD dataset. Figure 4.8A shows types of interactions existing in the PPI in-house dataset. Nearly all interactions are associated with the most generic "direct interaction" term, showing negligible representation of the remaining child terms. Figure 4.8 shows the same type of information but for protein interactions between the ADHD-associated proteins. Among identified interactions, there are 9 interactions of more specific type than the general "direct interaction", of which 4 are self-interactions. These are 4 "phosphorylation reactions", 4 "dephosphorylation reactions" and 1 "disulfide bond". The subset of ADHD-associated interactions demonstrates a similarly predominant level of generality in information content describing individual interactions as in the full PPI dataset. As such, the PPI datasets can be classified as a preliminary sieve.

Figure 4.9 shows the top 50 interaction detection methods of 125 terms associated with the full PPI dataset. Figure 4.10 shows all 47 interaction detection methods associated with the ADHD-associated PPI dataset. Both figures present a similar domination of two types of experiments, the "two hybrid" and the "pull down". The "two hybrid" method is a variant of transcriptional complementation assays. The "pull down" method belongs to affinity chromatography technology generally classified to biochemical methods.

The binary set of ADHD protein interactions is represented as a network graph. The network consists of 270 nodes (proteins) and 371 connecting them edges (interactions). After removing self-interacting nodes, the network consists of 243 nodes connected with 270 edges. As mentioned in Section 4.3.1, genes derived from the ADHdgene database have associated two levels of

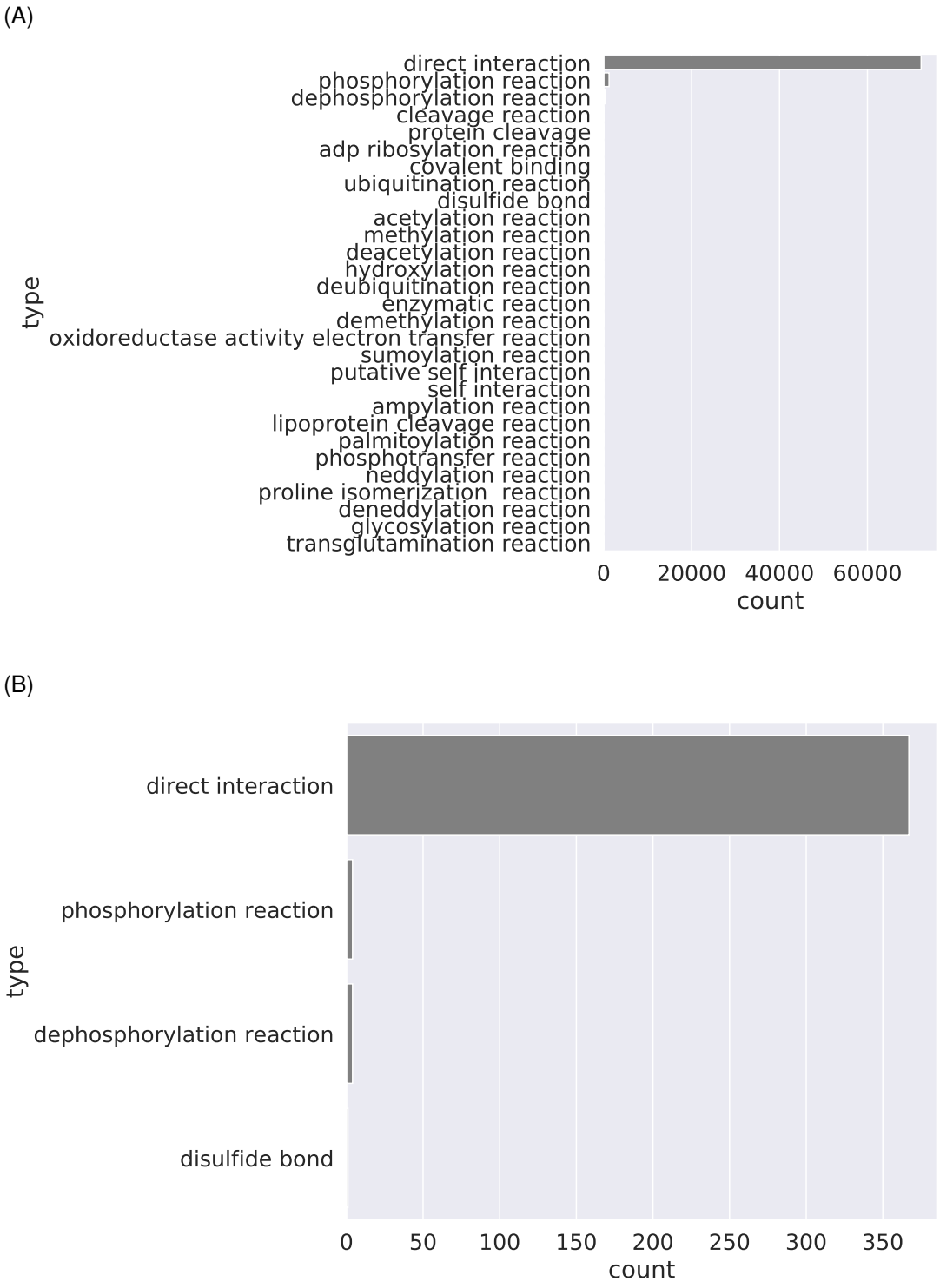


Figure 4.8: Distribution of "direct interaction" terms allocated per interaction for (A) all PPIs (29 terms); (B) PPIs for ADHD associated genes (4 terms). Datasets include duplicates of interactions if an interaction was assigned with different identifiers for interaction type. For ADHD associated genes, the number of interaction entries is 376, of which 371 are unique interactions.

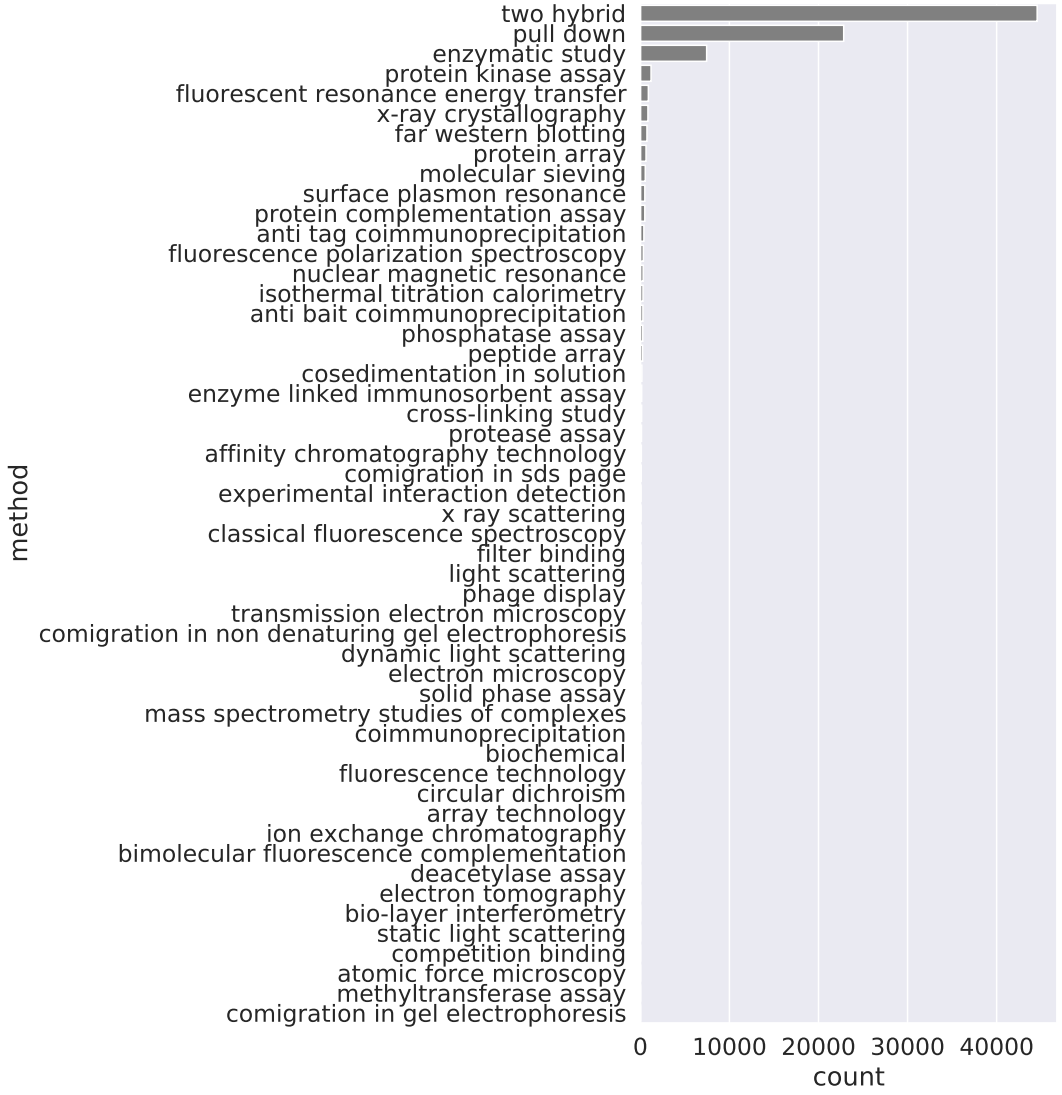


FIGURE 4.9: Distribution of the top 50 terms of the total 125 terms in the set of "experimental interaction detection methods" of the whole PPI dataset. Counts include duplicated interactions if an interaction was assigned with more than one interaction detection method.

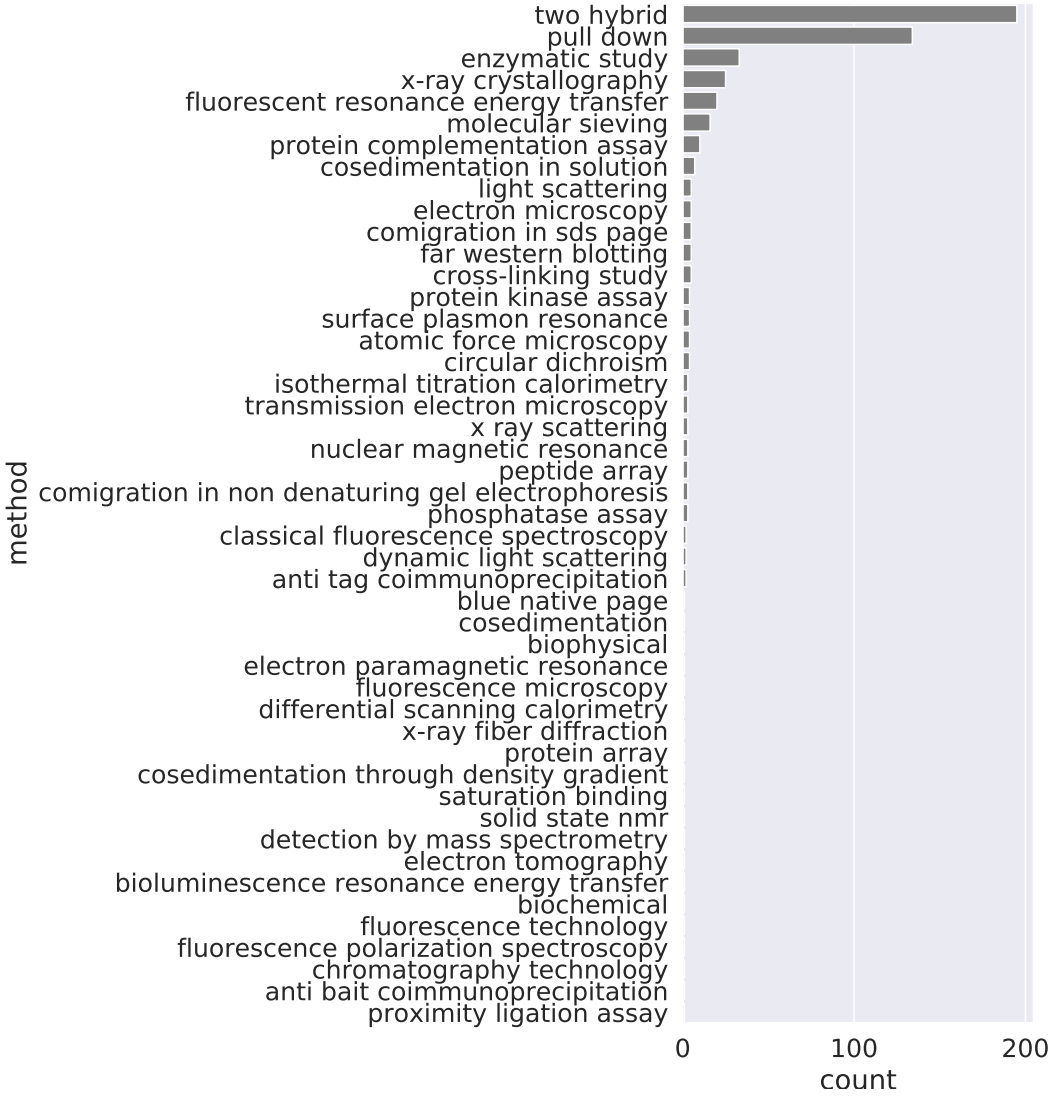


Figure 4.10: Distribution of the terms in the set of “experimental interaction detection methods” in the ADHD protein list (47 terms). Counts include duplicated interactions if an interaction was assigned with more than one interaction detection method. The number of data entries is 528 of which 371 are unique interactions. On average, there are 1.42 interaction detection methods assigned per each unique interaction. However, of 371 unique interactions, 295 (79.5%) are associated to only one method. A closer look at this subset shows that 150 interactions (40.4%) are confirmed solely by “two hybrid” and 87 interactions (23.5%) by “pull down” detection methods.

confidence, based on the number of studies that reported a significant gene-to-disorder relation. There are 38 genes confirmed by > 2 significant studies and 137 genes by > 1 significant study. The former 38 genes are mapped to the same number of proteins and 15 are found in the node list of the ADHD-associated PPI network. Of 137 genes, 135 are mapped to 145 proteins and 40 of them appeared in the node list.

Figures 4.11 and 4.12 present two perspectives of the network connectivity with removed self-interacting nodes for visualisation purpose. Node sizes in Figure 4.11 are proportional to the eigenvalue centrality scores. This centrality measure exposes nodes that are connected to a group of important nodes as the score is measured with respect to importance of neighbouring nodes determined by their degrees. Figure 4.12 demonstrates the location of cliques in the network. The minimal number of fully connected nodes is 3 with the maximal number reaching only 4. The network has few disconnected pairs of nodes and sparsely connected number of edges. This explains a still relatively high number of clusters that from 60 dropped to 33 after excluding self-interactions.

4.5.2 Protein-domain interaction network

Direct protein interaction networks presents rather sparse connectivity between the ADHD-associated proteins. They also does not offer a higher resolution view on interaction details, nor protein interaction interfaces, such as protein subunits that could hint on protein binding patterns. To include this information, proteins associated with ADHD are enriched with information on protein subunits collected from the Pfam database. Identified subunits of ADHD proteins are used to screen IDDI for protein-domain interactions. The obtained list of interacting domains is mapped back to proteins associated to ADHD. A new network is build with multiple edges denoting presence of interactions between protein components. It is important to note that the network represents a hypothetical view on interaction capabilities existing between proteins based on their characteristic subunits.

Table 4.4 reports the coverage of Pfam accessions and domain interactions with respect to the Human reference proteome and the ADHD proteins. Counts of DDIs are presented for a full set of DDIs and a subset of DDIs confirmed by at least one of the GSDDI datasets. Among domain interactions of

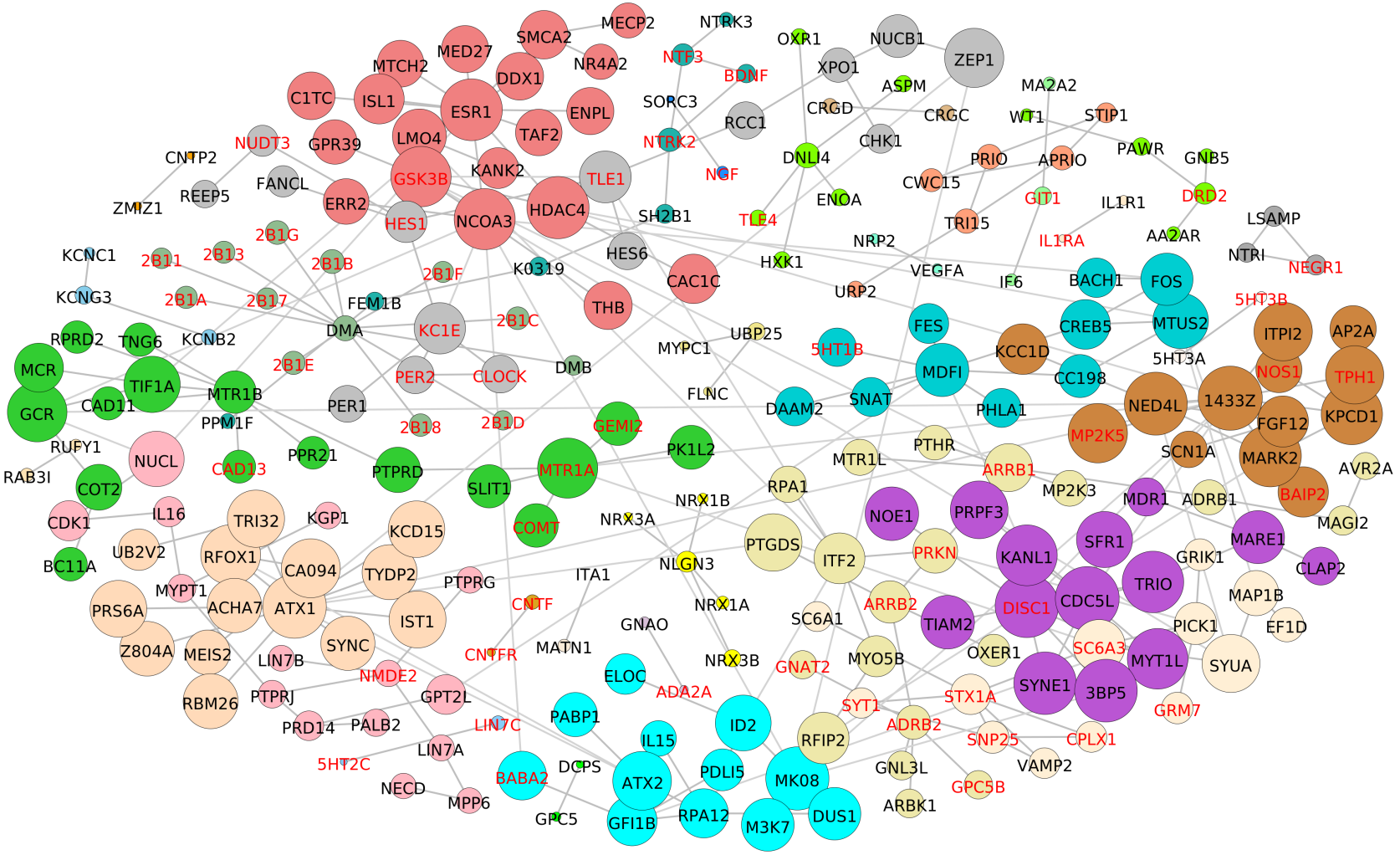


FIGURE 4.11: Centrality-view on results of clustering with multilevel algorithm on the 243 ADHD associated proteins connected with 270 edges. Node sizes are proportional to their eigenvalue centrality measures. Clusters are differentiated with colours. Nodes are labelled with UniProtKB entry names without indication of taxonomy for easier readability (e.g. CAC1S_HUMAN).

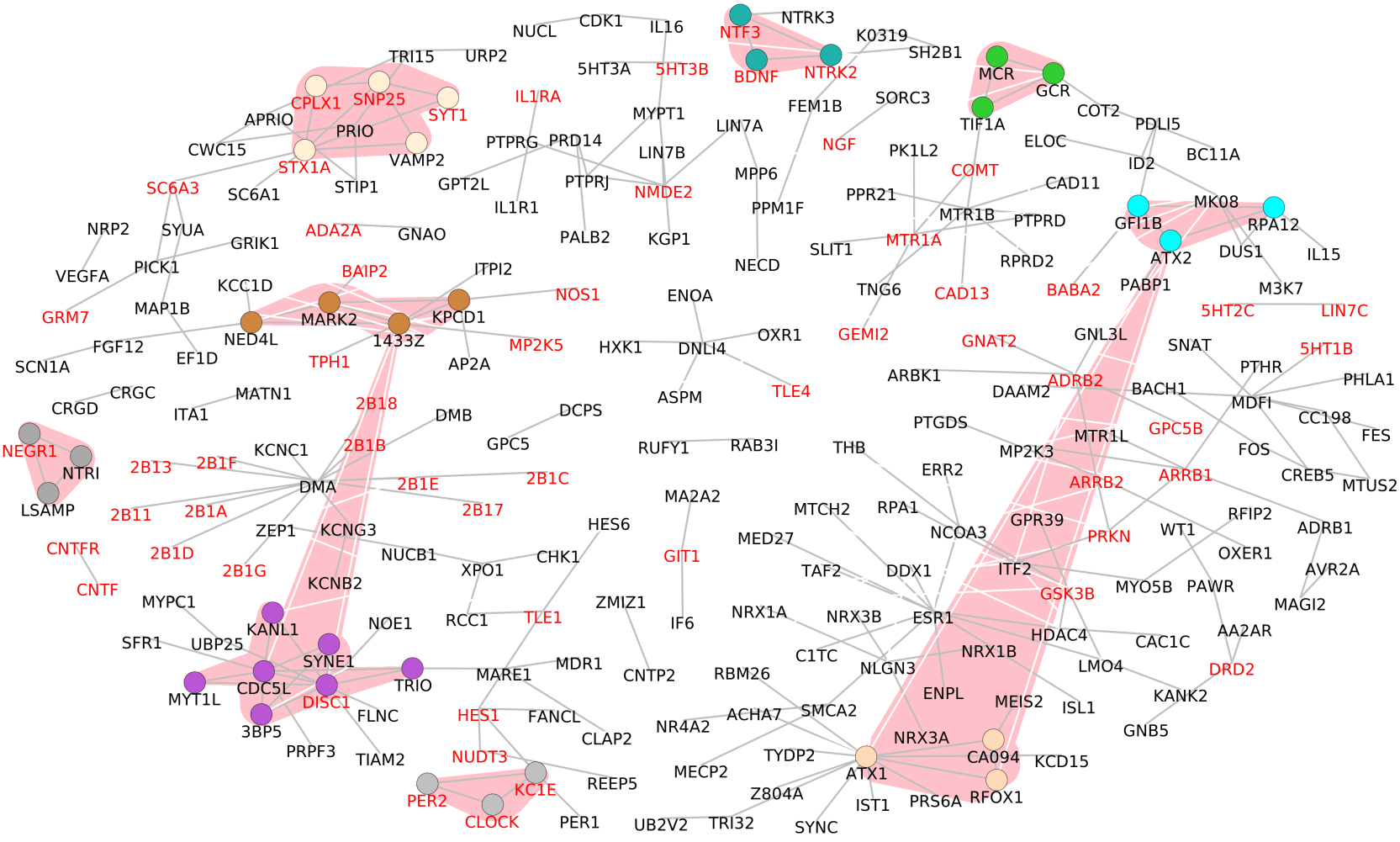


FIGURE 4.12: Clique-view on clustering results of 243 ADHD associated proteins with the multilevel algorithm. Cliques with at least 3 edges are overlaid onto cluster subgraphs presented in FIGURE 4.11 by using the same cluster coloration. Nodes are labelled with UniProtKB entry names without indication of taxonomy for easier readability (e.g. CAC1S_HUMAN).

Data types	Proteome Counts	ADHD proteins Counts
Genes	–	886
Proteins	70 891	760
Proteins with Pfam Acc	49 717	723
Pfam Acc mapped to proteins	6 116	723
Protein to Pfam Acc pairs	98 236	2 065
Domain interactions		
Domain interactions	130 843	11 294
Pfam Acc	4 271	595
Proteins	–	666
Domain interactions in GSDDI*		
Domain interactions	7 461	1 147
Pfam Acc	3 353	523
Proteins	–	623

TABLE 4.4: Counts of Pfam signatures and Human-specific domain interactions with respect to the Human reference proteome (2016_10) and the ADHD associated genes. *Interactions confirmed by at least one GSDDI.

the Human proteome, 5.7% is confirmed by at least one of the three GSDDI, whereas among the ADHD protein set, it is 10%. Nearly doubled domain interaction coverage can be explained by the character of ADHD-associated genes, that are often more studied than the average protein of the Human proteome. Moreover, the genes are mapped to canonical reviewed UniProtKB ACs (Swiss-Prot), whereas proteome is composed of both types of accessions, TrEMBL and Swiss-Prot.

A protein identifier can be mapped to more than one Pfam accession. In consequence, the number of protein interactions on the domain-resolution for ADHD grows combinatorially to 368117 potential protein-domain interactions. If only GSDDI are included, the number of hypothetical protein-domain interactions is reduced to 65196. An example of such combinatorial growth in domain interactions are the ones between "CUB and sushi domain-containing protein 3" (Q7Z407) and "CUB and sushi domain-containing protein 2" (Q7Z408). These proteins are connected with 1680 edges denoting domain

interactions. The first one has 42 domains, the second has 40 domains. As their names suggest, these proteins are composed of repeats of two the same domains, Sushi domain (PF00084) and CUB domain (PF00431). Both domains not only interact with each other but also interact with themselves. Therefore, the number of all possible combinations of interacts is equal to 1680 (40×42).

With domain interactions mapped to proteins, a network of potential protein interactions is constructed. The network can be viewed with two different levels of stringency, dependent on the included resources of DDIs, as a full set of DDIs or confirmed by at least one of the GSDDI databases. The network is clustered with the same "multi-level" algorithm used for the PPI dataset. The algorithm partitioned the network to 3 and 33 communities for the DDI- and the GSDDI-based networks, respectively. As the number of vertices and edges in these networks render visualisation difficult to analyse, Figure 4.13 shows the most numerous cluster (142 proteins) among the 33 communities that demonstrates various density regions.

Table 4.5 collates counts of the two networks. Networks are processed in two ways, by removing isolated nodes and by merging multiple edges into one between pairs of nodes and preserving the removed number of edges as an edge weight. Alongside the number of edges and nodes, the change in coverage of the 38- and 137-seed genes are reported. The drop in the seed gene coverage and the number of protein nodes between DDI and GSDDI is rather low. This allows to first concentrate on the GSDDI-based dataset in the next steps of the network reconstruction. The argument against the use of less stringent DDI dataset is also the lack of certainty in correctness of prediction-based sources of DDI.

Domain-based protein interaction networks includes far more protein nodes in comparison with the PPI-based network composed of 243 nodes. It is true even with the most stringent network variant. We can compare the number of edges between the DDI-based and the PPI-based networks after edge simplification. The edge simplification involves replacing multiple edges of the preliminary domain-based network with a single edge between a node pair weighted with the sum of multiple edges. The domain-based network has still much larger number of edges (9685) than the protein-interaction-based (270). The number of edges, being hypothetical interactions in the protein-based PPI network, increased 36 folds alongside the number of interacting

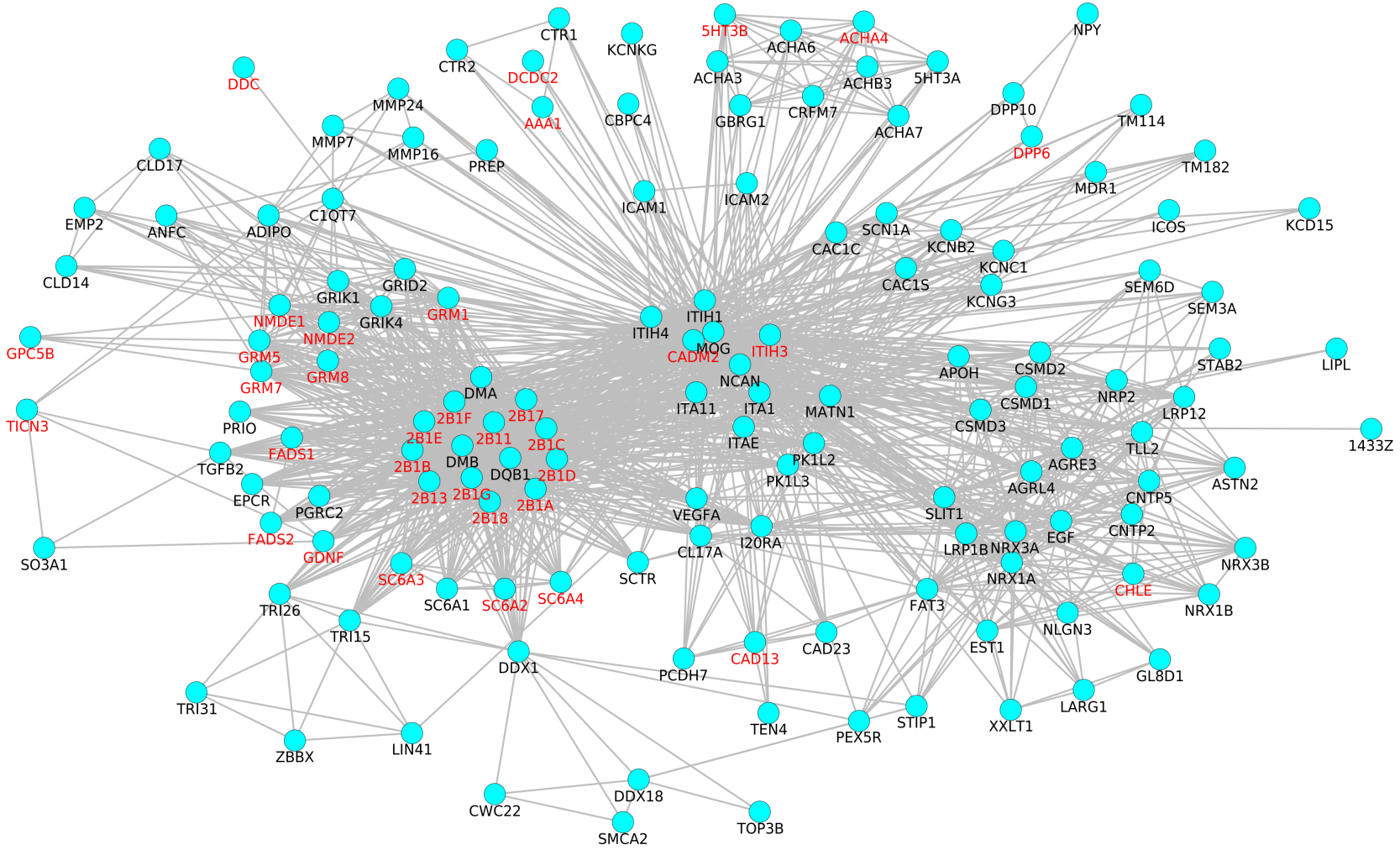


FIGURE 4.13: The largest cluster with 142 proteins in the clustering of 545 ADHD associated proteins with domain interactions confirmed by at least one GSDDI. Node labels are UniProtKB entry names with removed indication of taxonomy for easier readability (e.g. CAC1S_HUMAN). Red labelled proteins belong to the subset of 137 ADHD genes confirmed by at least one significant study.

Data types	DDIs Counts	GSDDIs Counts
Preliminary network		
Protein nodes	666	623
38-seed nodes	38	37
137-seed nodes	136	131
Edges	368 117	65 196
Merged edges	62 253	9 685
Isolated nodes	13	78
Network without isolated nodes		
Protein nodes	653	545
38-seed nodes	38	33
137-seed nodes	133	116
Merged edges	62 253	9 685
Clusters	3	33

TABLE 4.5: Counts of network components represented for two categories of stringency in separate columns: all sources of DDIs and interactions confirmed by at least one GSDDI. The network is processed in two ways, by merging multiple edges between node pairs into one and by removing isolated nodes.

proteins included in the network. Distribution of weights per edge for the ADHD-associated PDI network limited to GSDDI is visualised in Figure 4.14. For the network including all IDDI interaction sources in Figure 4.15. In both cases, the most common number of interacting sites is 1 with the most extreme weights reaching 1764.

4.5.3 Kinase-substrate interaction network

PhosphoSitePlus® is used as an exemplary resource of kinase-substrate interactions (KSIs). These dataset has not been presented in the previous sections as this dataset was used in its original form. The data were extracted from the PhosphoSitePlus® database[21] and screened with the ADHD genes. Of the total 17255 KSIs reported by the database, 10188 have both Human interactor proteins. In this Human-specific subset, 102 interactions between 45 ADHD-associated proteins were found. Neither of them is in the set of significant ADHD genes (Figure 4.16). The network has 14 genes that are kinases and 45 are substrates. This means that these 14 play both roles. This mechanisms is not a surprise as many substrates are themselves kinases. Similarly, many proteins are auto-phosphorylated.

The kinase-substrate pairs are screened unilaterally, by either mapping the ADHD genes to substrates or kinases. There are 121 ADHD genes that are substrates among the 547 kinase-substrate pairs, where kinases are outside of the ADHD gene set. On the other site, 19 ADHD proteins are involved in 1283 substrate-kinase interactions, when substrates are outside of the ADHD gene list.

4.5.4 Proteins in models

A common approach to parametrisation of new models is to use kinetic rates from previously modelled systems. To find out if any of the ADHD genes have been included in modelled systems, the proteins were screened through the resources of the BioModels database. The database, as oppose to CellML or DOQCS model repositories, offers unprecedented cross-referencing with other common biological collections of data and therefore, is used in this study.

Of 760 ADHD proteins, 40 were modelled in at least one model and 13 of them belong to the gene set confirmed by at least one significant study (the

[21] Accessed 2018-01-20.

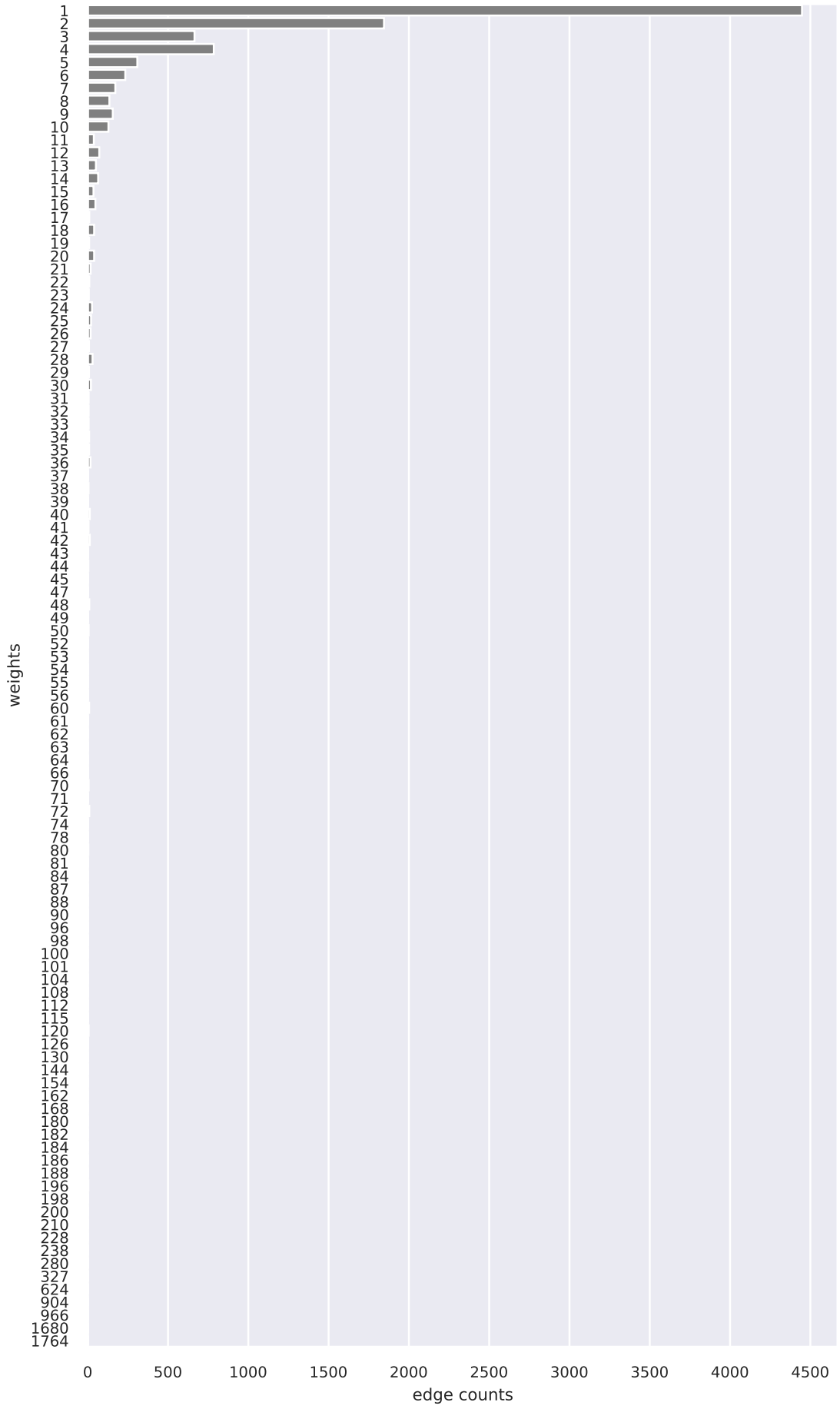


Figure 4.14: Distribution of edge weights of the ADHD-associated PDI network limited to GSDDI.

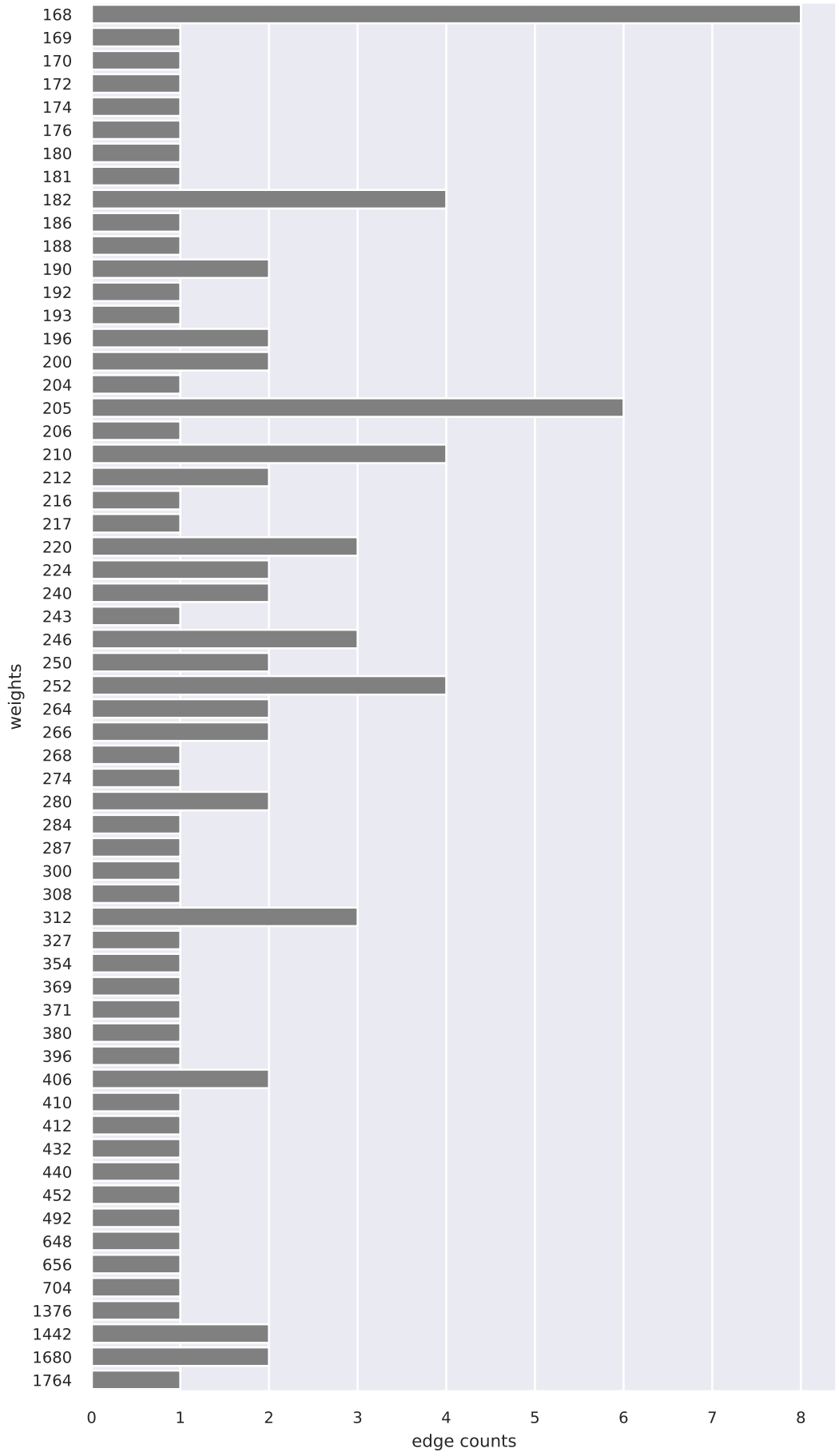


Figure 4.15: Distribution of the top 60 edge weights of the ADHD-associated PDI network with all DDIs included.

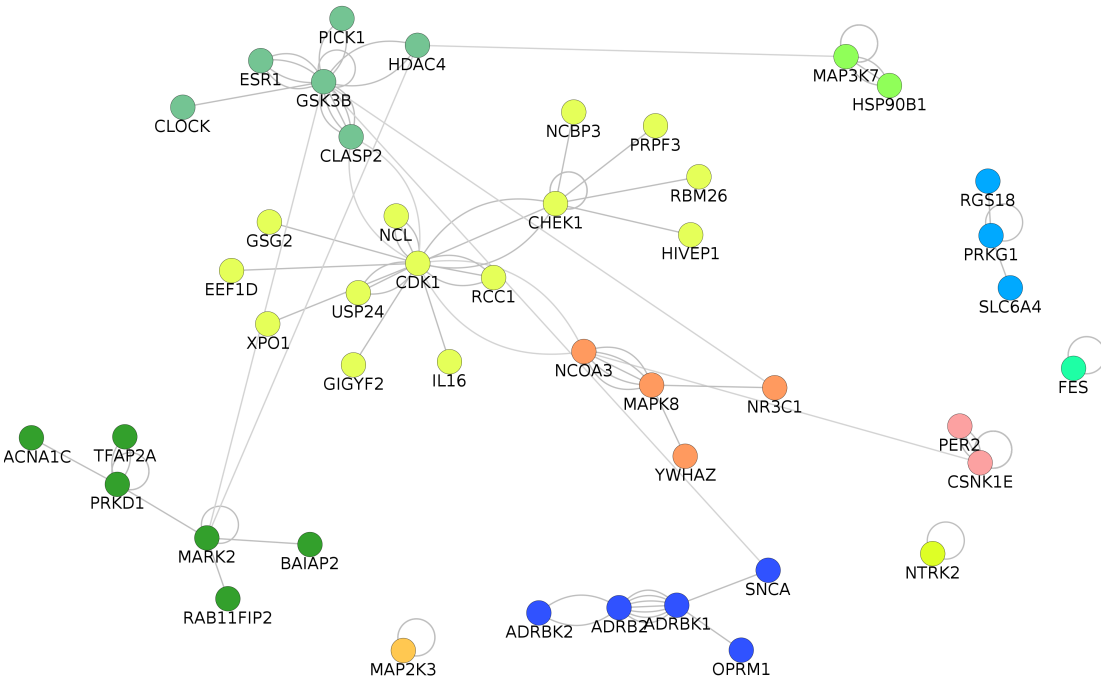


FIGURE 4.16: The kinase-substrate network of the ADHD-associated genes. The network is composed of 102 interactions between 45 ADHD-associated genes. Neither of the proteins belong to the set of genes confirmed by at least one significant study (the 137-gene set). The network is clustered with the "multi-level" algorithm. Clusters are designated by distinctive node colours.

137-gene set). There are 72 models with at least one ADHD protein. However, model identifiers point to 56 unique publications meaning that there are more than one model reported by a single publication. These models represent different variants or conditions of the same molecular system and were separately submitted to the BioModels database. Among 72 retrieved models, some are associated with multiple ADHD proteins. There are maximally six proteins that were modelled together in a single publication of the α-synuclein-based model of Parkinson's disease by Sass et al. [381]. The top modelled protein, Pro-epidermal growth factor (P01133), appeared in 9 publications and 11 models of the epidermal growth factor receptor (EGFR) signalling. Of the 56 publications, 38 modelled exactly one ADHD protein. Table B.1 in *Appendix B* enlists ADHD proteins with names of models they were found in.

4.5.5 Pathway enrichment analysis

Pathways represent grounded information about molecular mechanisms. Though limited in coverage, pathway information can serve as a basis of model development. In this section, the ADHD associated genes are subjected to pathway-based enrichment analysis, in particular its variant of ORA (Section 1.3.2). ORA is preformed on pathway data retrieved from the REACTOME database with the "`topONTO`" package (Section 4.4.3). Obtained p-values were corrected for the false discovery rate with the Benjamini and Yekutieli multiple testing correction [378]. Among the total of 886 ADHD genes, 439 are annotated to REACTOME terms. Table 4.6 shows enriched pathway terms in the ADHD-associated genes that scored with p-values < 0.01, after the correction was applied. Before the correction, 33 terms were found with p-values < 0.01. Terms with at least one annotated gene found among the ADHD-associated genes were included in calculations.

Figure 4.17 shows a hierarchical representation of reduced ontology graph with the significantly enriched nodes in the ADHD-associated genes. All 9 terms listed in Table 4.6 are leaf-level terms, as intended by application of the "`topONTO`" package. Found pathways are biologically related and centred around processes involving three neurotransmitters: serotonin, norepinephrine and dopamine. The last two neurotransmitters are catecholamines. In fact, norepinephrine is derived from dopamine, and their synthesis can be found among enriched pathway terms ("Catecholamine biosynthesis"). The

	REACTOME.ID	Term	Level	Annotated	Significant	Expected	elimFisher	elimFisherBY
1	R-HSA-390666	Serotonin receptors	7	12	11	0.52	1.14e-14	1.56e-10
2	R-HSA-390696	Adrenoceptors	7	8	8	0.35	1.25e-11	8.61e-08
3	R-HSA-212676	Dopamine Neurotransmitter Release Cycle	5	22	10	0.96	9.26e-09	4.25e-05
4	R-HSA-209931	Serotonin and melatonin biosynthesis	5	5	5	0.22	1.56e-07	4.29e-04
5	R-HSA-390651	Dopamine receptors	7	5	5	0.22	1.56e-07	4.29e-04
6	R-HSA-209905	Catecholamine biosynthesis	5	4	4	0.17	3.60e-06	6.63e-03
7	R-HSA-418555	G alpha (s) signalling events	5	143	20	6.25	3.76e-06	6.63e-03
8	R-HSA-181429	Serotonin Neurotransmitter Release Cycle	5	17	7	0.74	3.86e-06	6.63e-03
9	R-HSA-181430	Norepinephrine Neurotransmitter Release Cycle	5	18	7	0.79	6.08e-06	9.28e-03

TABLE 4.6: Results of pathway enrichment analysis with the REACTOME dataset, performed with the "`topONTO`" package. A Human genome is selected as a background dataset. A set of genes of interest consists of the full set of genes associated to ADHD. Column names: *REACTOME.ID* – REACTOME identifier; *Term* – full pathway name; *Level* – position of the term in the ontology graph; *Annotated* – total number of genes annotated to the term; *Significant* – the number of genes of interest found to be associate to the term; *Expected* – the number of genes to be associated by chance; *elimFisher* – the p-value score of the Fisher test for "`elim`"; *elimFisherBY* – the p-value score of the Fisher test for "`elim`" corrected with the Benjamini and Yekutieli method.

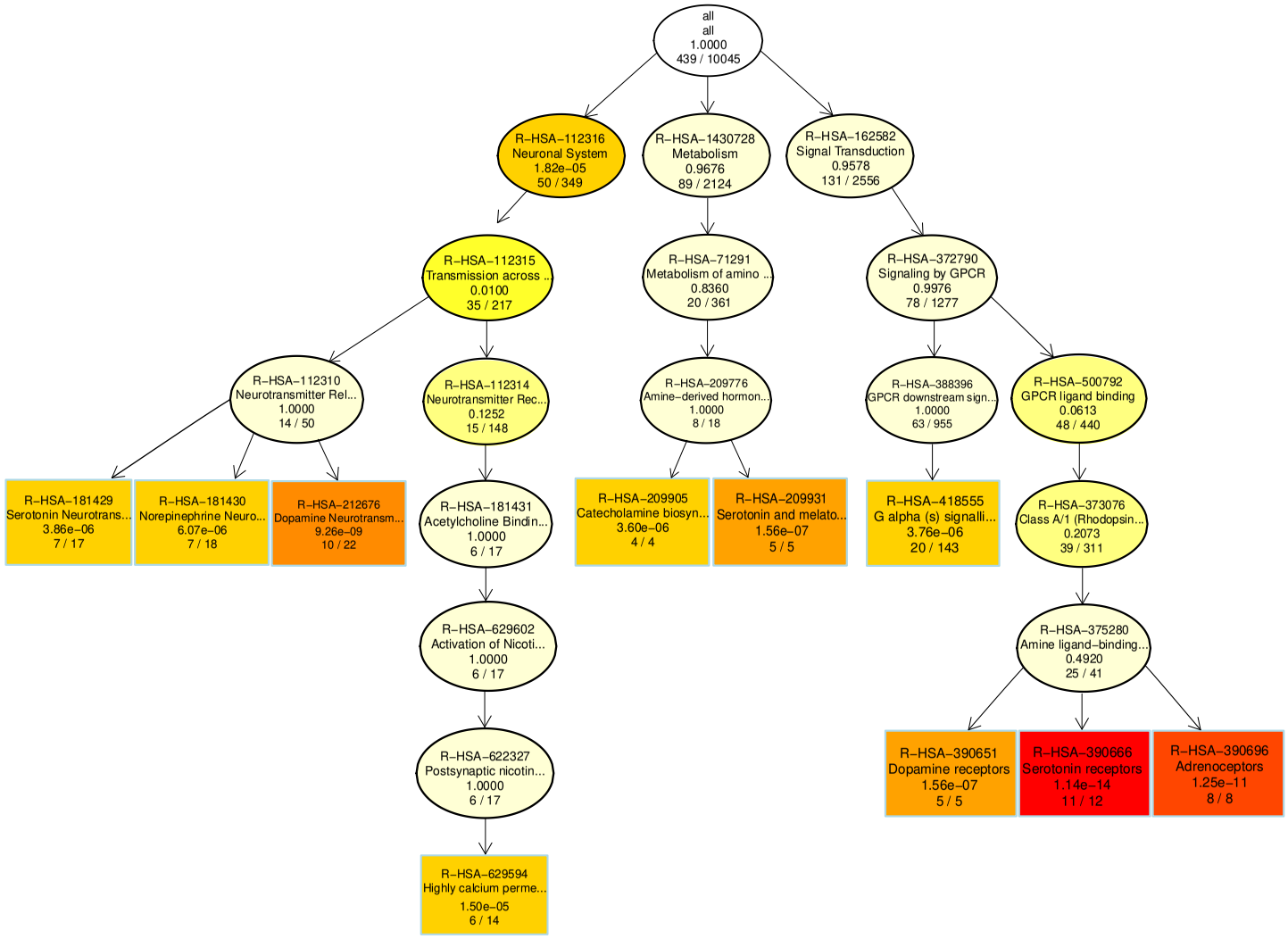


FIGURE 4.17: Hierarchical graph representation of pathway enrichment analysis with REACTOME. Significant terms with p-value below ≤ 0.01 before Benjamini and Yekuteili corrections are in rectangular nodes. Nodes are coloured with the scale from yellow to red for least significant to most significant, respectively. The Human genome is selected as a background dataset, and a set of genes of interest is composed of the full set of genes associated to ADHD.

other enriched pathways, localised in the presynaptic terminal, are neurotransmitter release cycles to synaptic cleft. Lastly, pathways involving receptors that are targeted on the postsynaptic membranes by the three neurotransmitters are enriched.

Sharing of genes between pathways can be observed in FIGURE 4.18, depicting a network of pathway terms connected to their gene members, represented as symbols. Pathway identifiers are marked in blue, whereas the subset of genes associated to ADHD, in red. The network is clustered with the "`multi-level`" algorithm. Clusters are designated by distinctive node colours. The largest green cluster concentrates terms around the "G alpha (s) signalling events" pathway (`R-HSA-418555`), which shares genes with "Dopamine receptors" (`R-HSA-390651`), "Adrenoceptors" (`R-HSA-390696`) and "Serotonin receptors" (`R-HSA-390666`). The most entangled pathways regarding shared genes are "Dopamine Neurotransmitter Release Cycle" (`R-HSA-212676`), "Serotonin Neurotransmitter Release Cycle" (`R-HSA-181429`) and "Norepinephrine Neurotransmitter Release Cycle" (`R-HSA-181430`) (bright green cluster).

4.5.6 Protein interactions enhanced with domain information

If self-interactions are excluded, the PPI network is composed of 243 nodes and 271 edges, whereas the domain-based interaction network has 545 proteins and 9685 edges. This large increase of potential interactions hidden in the domain-level, might not correspond to the actual interactions. The domain-level interactions does not include structural constraints of interaction interfaces that might render two domains as non-interacting. On the other hand, experimentally confirmed protein interactions have various level of certainty regarding limitations of detection methods. Majority of protein interactions is detected with the "two hybrid" method that is known to have a high false-positive rate [382]. Combined information from PPI and DDI datasets can limit protein interactions to the ones of higher confidence, though with the risk of false negatives, as protein domains are not the only means of protein interactions.

The number of PPI, including self-interactions (271 nodes, 371 edges), that has at least one domain-domain interaction is 165 (44.5%). In a network without self-interactions (243 nodes, 271 edges), 69 (25.5%) interactions be-

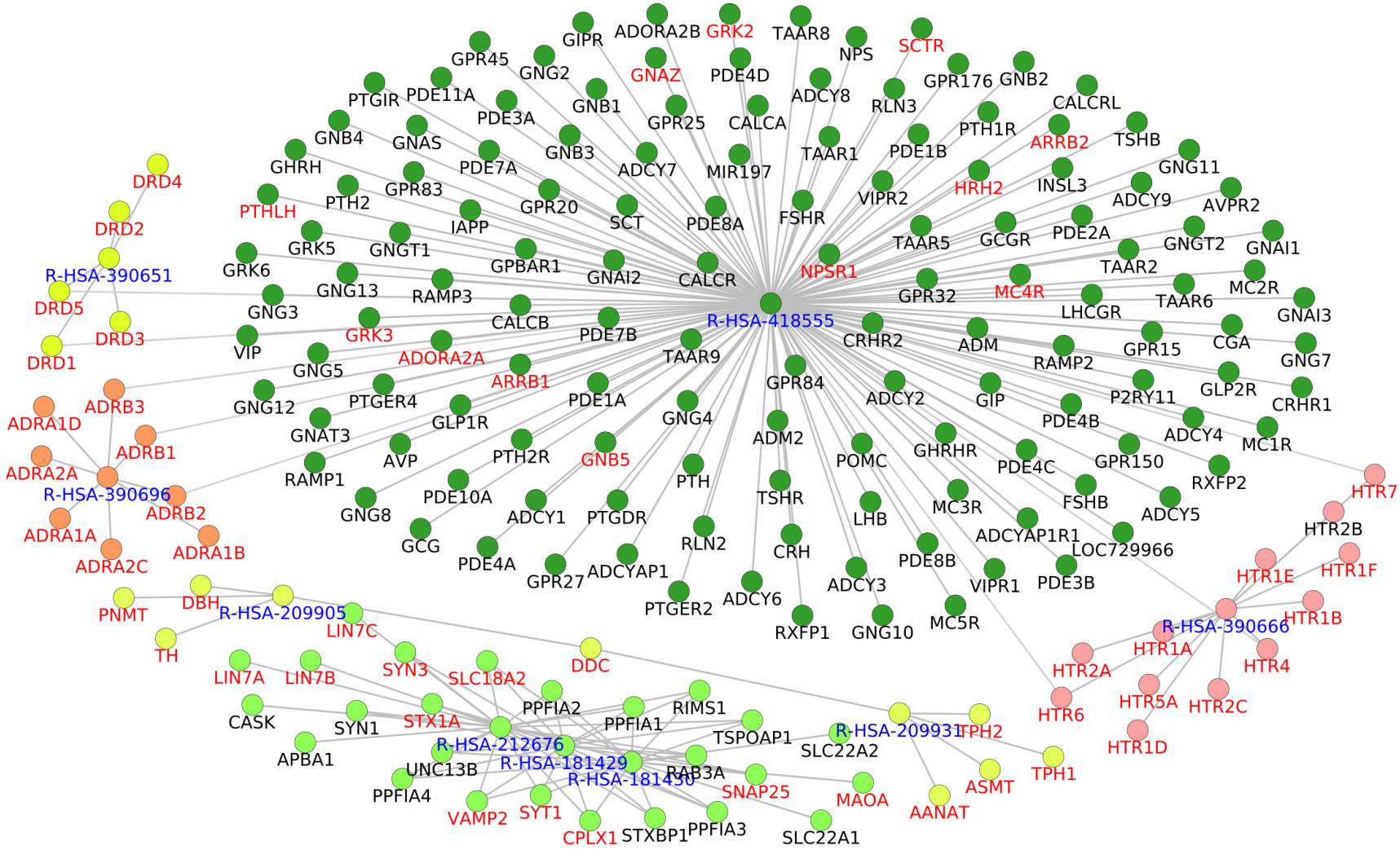


FIGURE 4.18: Network representation of 9 enriched pathways that include all pathway members denoted with gene symbols. Red-node labels denote the ADHD-associated genes found in pathways. Blue-node labels denote pathway REACTOME identifiers. The network is clustered with the "`multi-level`" algorithm. Clusters are designated by distinctive node colours.

	REACTOME.ID	Term	Level	Annotated	Significant	Expected	elimFisher	elimFisherBY
	"V1"							
1	R-HSA-264642	Acetylcholine Neurotransmitter Release Cycle	5	17	5	0.01	7.26e-15	4.61e-11
2	R-HSA-181429	Serotonin Neurotransmitter Release Cycle	5	17	5	0.01	7.27e-15	4.61e-11
3	R-HSA-181430	Norepinephrine Neurotransmitter Release Cycle	5	18	5	0.01	1.00e-14	4.61e-11
4	R-HSA-888590	GABA synthesis, release, reuptake and degradation	5	19	5	0.01	1.37e-14	4.70e-11
5	R-HSA-212676	Dopamine Neurotransmitter Release Cycle	5	22	5	0.01	3.09e-14	8.51e-11
6	R-HSA-210500	Glutamate Neurotransmitter Release Cycle	5	24	5	0.01	4.99e-14	1.14e-10
7	R-HSA-5250958	Toxicity of botulinum toxin type B	6	3	2	0	5.95e-07	9.09e-04
8	R-HSA-5250989	Toxicity of botulinum toxin type G	6	3	2	0	5.95e-07	9.09e-04
9	R-HSA-5250971	Toxicity of botulinum toxin type C	6	3	2	0	5.95e-07	9.09e-04
10	R-HSA-422356	Regulation of insulin secretion	4	80	3	0.04	4.81e-06	6.61e-03
	"V18"							
1	R-HSA-383280	Nuclear Receptor transcription pathway	4	51	5	0.04	5.75e-11	7.91e-07
	"V27"							
1	R-HSA-1296072	Voltage gated Potassium channels	4	44	3	0.01	7.84e-08	1.08e-03

TABLE 4.7: Results of pathway enrichment analysis with REACTOME obtained with the `topONTO` package. A set of 28 clusters of ADHD proteins derived from protein interaction network that at least have one pair of domains is tested against a human genome as a background data set. Three clusters named "V1", "V18" and "V27" resulted with at least one pathway. The "V18" cluster has 7 members, the "V27" cluster has 3 members and the "V1" cluster has 5 members. Column names: *REACTOME.ID* – REACTOME identifier; *Term* – full pathway name; *Level* – position of the term in the ontology graph; *Annotated* – total number of genes annotated to the term; *Significant* – the number of genes of interest found to be associate to the term; *Expected* – the number of genes to be associated by chance; *elimFisher* – the p-value score of the Fisher test for "`elim`"; *elimFisherBY* – the p-value score of the Fisher test for "`elim`" corrected with the Benjamini&Yekutieli method.

tween 94 (38.7%) proteins has at least one domain-domain interaction. The network has 28 detached components that divide the network into clusters. To learn if any of these clusters represent a known biological pathway, 28 clusters are subjected to pathway enrichment analysis based on the pathway set deposited in REACTOME. Of 28, 3 rendered results that passed the threshold of p-value < 0.01, after the Benjamini and Yekutieli multiple testing correction. Two clusters resulted with an enrichment of a single pathway. Both pathways have relatively large number of associated genes ("V18" and "V27" Table 4.7) and 4th level on the REACTOME pathway tree. The third "V1" cluster resulted with 10 pathways, with all 5 cluster members associated with the first 6 pathways located on the lower 5th level of the pathway diagram (Table 4.7). These 6 pathways are siblings, grouped under a common ancestor pathway term, "Neurotransmitter release cycle" (R-HSA-112310). Among pathways, not indicated in the enrichment of the whole ADHD-associated gene set (Table 4.6) are the "GABA synthesis, release, reuptake and degradation" pathway (one of the 6 sibling pathways), the "Regulation of insulin secretion" and three pathways of the toxicity of botulinum toxin, each of the three annotated with only 3 genes.

Figure 4.19 show network representation of pathways enlisted in the "V1" cluster in Table 4.7.

To examine if enrichment on the full set of 94 proteins results with a different set of enriched pathways, that is if division into clusters might occlude enrichment of other pathways, the list of 94 proteins is subjected to pathway enrichment. Table 4.8 presents results of the enrichment of the undivided protein set. The first 6 pathways are among 6 pathways in the "V1" cluster in Table 4.7. These are "GABA synthesis, release, reuptake and degradation" and 5 pathways of neurotransmitter release cycles of acetylcholine, serotonin, norepinephrine, dopamine and glutamate.

The "Nuclear Receptor transcription pathway", located at the bottom in Table 4.8 of enrichment results for the undivided protein list, can be found in the enrichment results of the "V18" cluster (Table 4.7). This pathway does not share any of its members with other enriched pathways (Figure 4.20). Pathway enrichment analysis of the unclustered protein indicated the "Adrenoceptors" pathway (R-HSA-390696) that was missed in the cluster-based enrichment. Similarly to the "Nuclear Receptor transcription pathway" (R-HSA-383280), the "Adrenoceptors" pathway (R-HSA-390696) members also form detached clus-

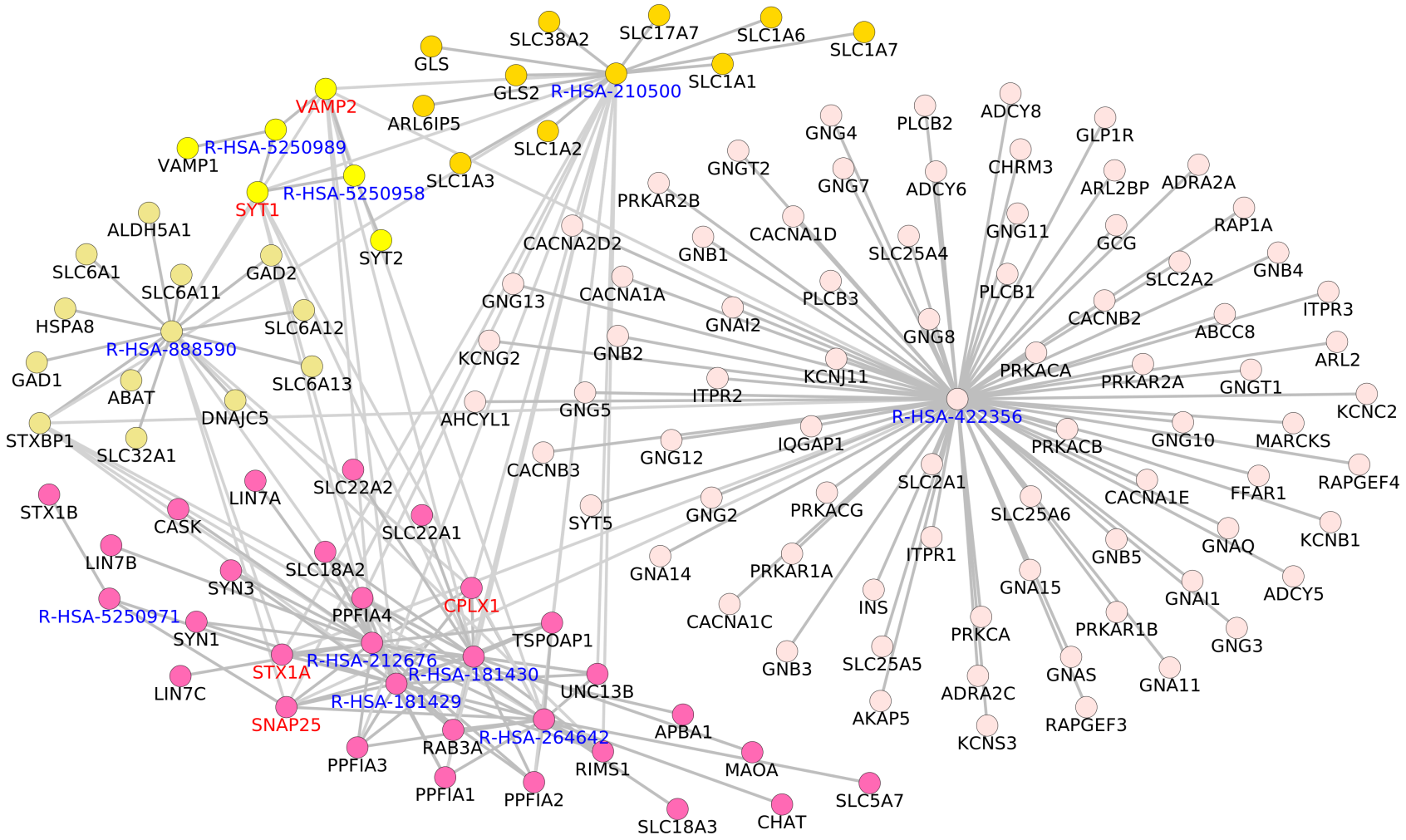


FIGURE 4.19: Network representation of 10 enriched pathways with the genes of the "V1" cluster. Red-node labels denote 5 ADHD genes of the cluster. Blue-node labels denote 10 pathway REACTOME identifiers. The remaining gene symbols are other gene members of the enriched pathways. The network is clustered with the "`multi-level`" algorithm. Clusters are designated by distinctive node colours.

	REACTOME.ID	Term	Level	Annotated	Significant	Expected	elimFisher	elimFisherBY
1	R-HSA-212676	Dopamine Neurotransmitter Release Cycle	5	22	7	0.14	4.81e-11	6.62e-07
2	R-HSA-181429	Serotonin Neurotransmitter Release Cycle	5	17	5	0.11	5.22e-08	2.40e-04
3	R-HSA-264642	Acetylcholine Neurotransmitter Release Cycle	5	17	5	0.11	5.22e-08	2.40e-04
4	R-HSA-181430	Norepinephrine Neurotransmitter Release Cycle	5	18	5	0.11	7.20e-08	2.48e-04
5	R-HSA-888590	GABA synthesis, release, reuptake and degradation	5	19	5	0.12	9.72e-08	2.67e-04
6	R-HSA-210500	Glutamate Neurotransmitter Release Cycle	5	24	5	0.15	3.47e-07	7.95e-04
7	R-HSA-390696	Adrenoceptors	7	8	3	0.05	1.35e-05	2.65e-02
8	R-HSA-383280	Nuclear Receptor transcription pathway	4	51	5	0.32	1.68e-05	2.89e-02

TABLE 4.8: Results of pathway enrichment analysis with REACTOME obtained with the `topONTO` package. A set of ADHD protein interaction network that have at least one pair of domains is tested against the Human genome as a background data set. Column names: *REACTOME.ID* – REACTOME identifier; *Term* – full pathway name; *Level* – position of the term in the ontology graph; *Annotated* – total number of genes annotated to the term; *Significant* – the number of genes of interest found to be associate to the term; *Expected* – the number of genes to be associated by chance; *elimFisher* – the p-value score of the Fisher test for "`elim`"; *elimFisherBY* – p-value score of the Fisher test for "`elim`" corrected with the Benjamini&Yekutieli method.

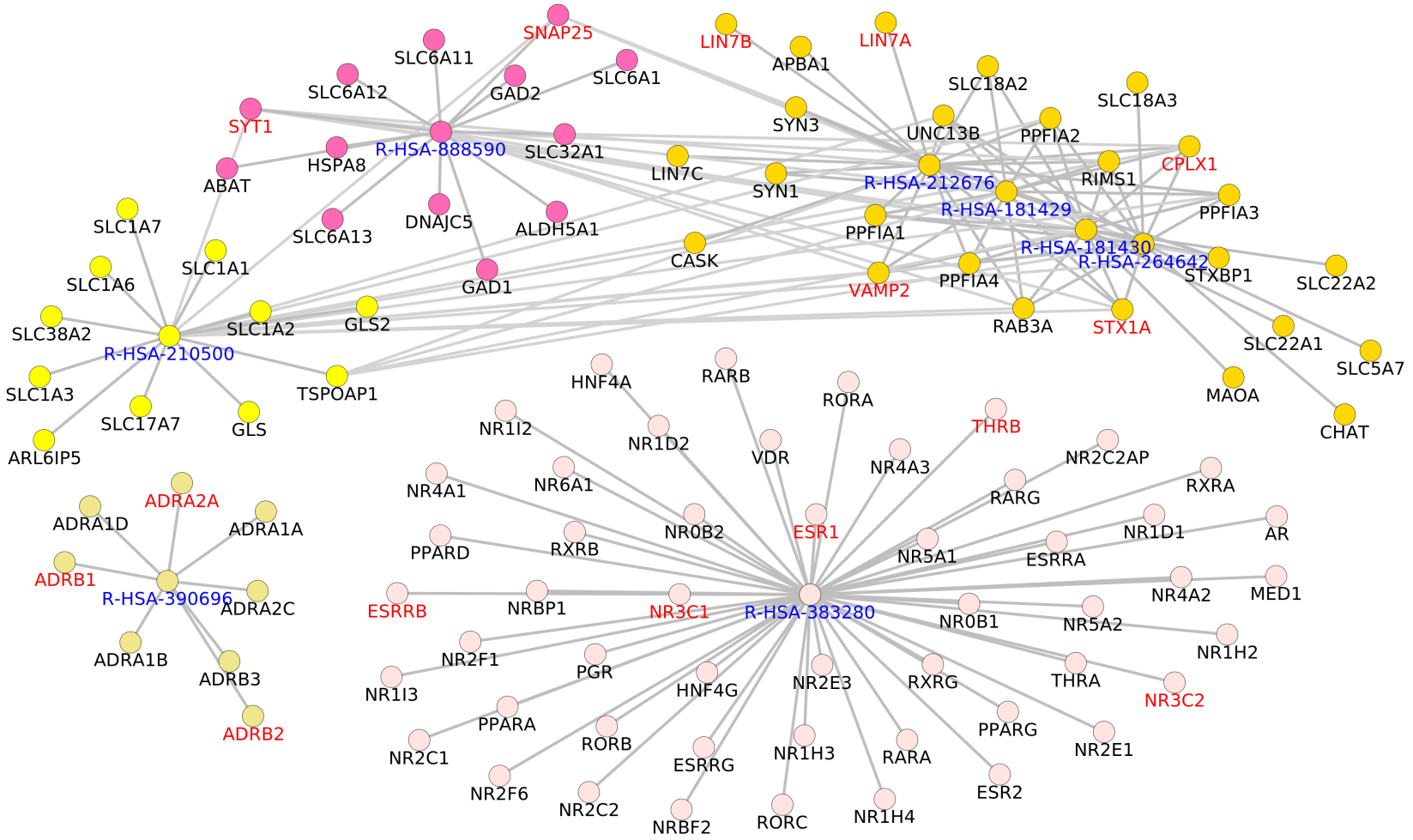


FIGURE 4.20: Network representation of 8 enriched pathways enlisted in TABLE 4.8. All pathway members are shown in the network. Red-node labels denote the ADHD-associated genes found in pathways that belong to the subset of these genes having known protein interactions and where at least one pair of interacting domains could be found. Blue-node labels denote pathway REACTOME identifiers. Black-labelled nodes are pathway members. The network is clustered with the "`multi-level`" algorithm. Node colours designate clusters.

ter in the network representation of enriched pathways (Figure 4.20). On the other hand, the cluster-based enrichment indicated the "Regulation of insulin secretion" pathway (R-HSA-422356) as significantly enriched, that is missed in the enrichment of unclustered proteins. In Figure 4.19, showing the network representation of enriched pathways of the "V1" cluster, enriched pathways share their members in larger degree, expected by the underlying protein interaction module.

4.6 Discussion

The realm of biological data is ever-changing environment that require the constant process of verification. There is a limited fraction of confirmed and stable datasets that forces ventures to include and integrate datasets obtained with high-throughput experimental methods and computational predictions. These, on the other hand, suffer from uncertainty, errors and redundancy. Operating between different data sources, implied by an integrative approach, introduces another layer of difficulties regarding the control for errors in datasets. Even process of merging resources of similarly understood biological entities, like domains and pathways, is problematic. However, ongoing community efforts and distinctive progress to tackle these issues is observed.

The aim of this chapter was to identify aspects and difficulties that assembling of a RB model from molecule-centred datasets might imply. To reflect a potential biomedical inquiry, this exploration sets off from a list of genes associated with an exemplary complex disorder, that is ADHD. The list of genes is assembled from three variable datasets that represent high-quality curated and automatically assembled resources. Different types of knowledge-bases and association datasets were employed to learn on biological meaning represented by the list of ADHD-associated genes.

The first association datasets are physical interactions between proteins. They are commonly used in molecular studies of diseases [383–387]. To limit interaction to direct and experimentally detected ones, an in-house dataset of PPIs was assembled by merge of three primary databases, standardised with the common PSI-MI data format. To select direct and experimentally detected protein interactions, PPIs with appropriate MI ontology terms were only preserved. However, as identified in the IntAct dataset, co-complex interactions, binarised to a tabular form with the spoke expansion model, are

in fact also categorised as "direct interactions". Given that only the IntAct database explicitly tags these interactions, classifying an interaction as direct that is reported by other databases, might introduce false positive records. In spite of this uncertainty, all three PPI resources were merged, preceded with removal of spoke-expanded interactions from IntAct. Overlap between the three datasets is negligible (0.67%). The overlap between two largest datasets (BioGRID and IntAct) is higher, though it is still quite low (14.69%). Exact mechanisms of interactions are not provided in PPI and therefore, using them alone will not allow to encode realistic interaction rules that emphasise binding interfaces. Therefore, another data types that were reached for were proteins annotated with domains and domain-domain interactions (DDIs). The protein-to-domain annotations were retrieved from the Pfam protein signature database. A network of PDIs was constructed from the updated IDDI database of DDIs composed of Human proteins. IDDI combines 23 datasets including 3 gold standard domain-domain interactions (GSDDIs). Mapping DDIs between ADHD genes revealed that there is 36-fold increase in the number of potential site-specific interactions (9685 edges) than found by mapping the same genes to the PPI dataset (271 edges). Furthermore, a much higher number of the ADHD-associated proteins was included in the PDI network than when PPIs were only considered. This domain-based protein interaction network reflect only potential interactions. A proof of the actual existence of these interactions would require further studies supported by other resources and statistical methods to eliminate conflicting domain interactions.

As PPIs might contain false positive records and a narrow overlap between databases was observed, the PPI network of the ADHD-associated proteins was intersected with DDIs. 38.7% of proteins of the ADHD-associated PPI network were found to be involved in at least one DDI (excluding self-interactions). The number of PPIs mediated by at least one DDI were 25.5% of the same PPI network of the ADHD-associated proteins. As reactions potentially mediated by domains cover less than 40% of proteins identified as directly interacting, other resources should be also employed to gain greater coverage and understanding of reaction details.

The ADHD-associated gene set was also mapped to kinase-substrate interaction (KSI) obtained from the PhosphoSitePlus® database. This mapping resulted with a network of sparse connections that included most studied

proteins but non of the proteins that association to ADHD was confirmed by at least one significant study. However, interactions in this network could be directly encoded as rules of a dynamical model, as they represent direct physical interactions. Moreover, these reactions belong to a particular class that could be parametrised with generalised reaction constants for phosphorylation reactions. Different strategy could be acquired to slightly increase coverage in phosphorylation reactions of genes associated to ADHD. When only substrates were mapped to the ADHD-associated genes and kinase proteins were outside of this set, 13.6% of these genes were found among substrates of KSIs, compared to 5% when both interactores were contained in the ADHD gene set.

PhosphoSitePlus® is only an example of existing resources of kinase-substrate interactions. Other databases could be examined and potentially integrated to evaluate coverage of this type of interactions. For instance, the PhosphoNetworks dataset includes a large Human dataset of phosphorylation sites and kinase-substrate relations [338]. Hu et al. [338] integrated high-throughput phosphorylation data and provided a high-resolution phosphorylation dataset. However, the dataset has not been updated since 2014. To other potential sources of kinase-substrate interactions is the Eukaryotic Kinase and Phosphatase Database (EKPD) [388]. Similarity to the Pfam families, the database contains defined families of eukaryotic protein kinases and phosphatases, for which position-specific scoring system based on the Hidden Markov Models, the HMM profiles, were generated. Based on these profiles, a genome-wide identification of kinases and phosphatases can be achieved [388].

Enrichment analysis with respect to the REACTOME pathway terms was performed to identify pathways that could be used as a starting point to develop dynamic models of disorder-related mechanisms. The pathway enrichment analysis was performed with the "`topONTO`" package. Unlike usual ORA methods, the one implemented in "`topONTO`" takes into account ontology structure to produce enrichment results. Consequently, more specific but still significantly enriched pathway terms were identified.

It is worth noting that around a half of the ADHD-associated genes was found as significantly enriched in any pathway term. The highest number of genes allocated to the same pathway was 20, in the largest pathway of "G alpha (s) signalling events". In other pathways, this number was not larger than 7.

Pathways enriched in the ADHD-associated genes were identified as

closely related, located in pre- and postsynapses. These pathways involve processes of synthesis, release cycle and interactions with receptor targets of 5 major neurotransmitters. A visualisation of pathways as clustered network graphs allowed to easily observe extensive sharing of genes between three of these pathways. These pathways appeared to be sibling-terms on the ontology tree, parented by the more general pathway of neurotransmitter release cycle. Enrichment performed on proteins derived from the intersection of PPIs and DDIs of the ADHD-associated genes revealed three other pathway members of neurotransmitter release cycles that involve glutamate and acetylcholine release cycles, and the GABA synthesis, release and degradation pathway.

To convey any distinctive conclusions regarding the disease mechanisms from the performed analyses, the obtained results would have to be consulted with the specialist in the ADHD domain. Therefore, I refrain from a discussion on potential disease mechanisms that could possibly be derived from the identified pathways. However, it is worth noting that indication of neurotransmitter-related process by the enrichment analysis is cohesive with generally known symptoms of ADHD. Disruption of levels of multiple neurotransmitters in ADHD is commonly admitted [389]. Typically used pharmacotherapies involve medications that increase extra-synaptic levels of dopamine and norepinephrine (methylphenidate and amphetamine) [390]. Importance of GABAergic inhibitory neurons in the child ADHD, particularly in the upregulation of GABA inhibitory function, was experimentally evidenced by Nagamitsu et al. [391]. In other study by Edden et al. [392], reduced level of the GABA neurotransmitter concentration was observed in ADHD children compared to a control group.

Regarding the type of method for enrichment analysis that was used in this study, there are certain limitations in the ORA-based approach, in particular with respect to the pathway enrichment analysis. Firstly, pathways in the ontology-base ORA approach are considered as terms connected with each other to form parent-child relations. A parent is a more generic term that contains one or more children terms. Each term harbours a list of genes that are not connected within the list nor to other pathways. This representation neglects a vital information regarding pathway topology. Therefore, a potentially more adequate enrichment approach to pathway datasets is to combine the gene set-based scoring system with the pathway network topology. Methods

implementing such approach include information about a position of gene in a pathway topology diagram. For instance, CePa uses different centrality measures corresponding to diverse biological functions to weight pathway nodes found as differentially expressed genes [393]. Another proposition is EnrichNet [394] that overlays target genes and reference datasets onto a molecular interaction network combined with information about tissue-specific gene expression and computes network-based gene set enrichment scores.

Although very detailed, information enclosed in pathway datasets lack quantitative data characterising reactions they represent. There are multiple resources with quantitative data that are commonly used in modelling process. For instance, one of commonly used resources are public databases, such as SABIO-RK [345], BRENDA [395], IntEnz [396] and MetaCyc [397]. These databases are focused on enzyme kinetics, relevant for instance to protein kinases and phosphatases. Other quantitative information like protein concentration, binding affinities and reaction rates could be extracted from the literature either manually or by using natural language processing (NLP) techniques to find relevant publications and references in a high-throughput process of document screening [398]. If not from exact processes, quantitative measurements could be obtained from related processes. Other sources of kinetic information can be obtained from existing models published in repositories such as BioModels, CellML or DOQCS. These resources are rather scarce and limited to most popularly modelled molecular systems (e.g. EGFR) as seen on example of ADHD-associated proteins found in at list one model deposited in BioModels (5.3%). If quantitative data are nowhere to be found, a common approach is to create a parameter value window around default rates for particular classes of reactions [174].

An important source of information on how to automate the process of RB model construction are frameworks and infrastructures developed to address the ordinary differential equation (ODE) modelling, such as KiMoSys, a web-based repository for experimental data and model exchange [399]; SBMLsqueezer for semi-automated and integrative process of the ODE modelling that addresses assignment of kinetic rate laws, integrable with any modelling pipeline [111]; and Path2Models, a generator of kinetic, logic or constraint-based SBML-encoded models of metabolic processes from the KEGG pathway dataset [342].

Protein-centred associations, in form of PPIs, were used as an essential source of deriving functional links between disease genes in this study. There are other associations that could be used to relate genes and that have been used in disease studies, such as gene interactions, sample-specific gene transcripts and gene co-expression networks [400–402]. Genetic interactions point to functional relations between genes that affect each other and produce a phenotype that is not observed with their individual effects [403]. However, these interactions are most commonly studied with low-level model organisms (e.g. Fruit fly and Yeast) as genetics of these organisms can be easily manipulated, and effects of mutations are quickly observed due to these organisms short life cycles. The other mentioned data source are gene expression correlations. Pairs of differentially co-expressed genes are linked by virtue of similarity in their activity measured across multiple conditions [404]. Neither of these two datasets provide information on direct and causative relations between these proteins or genes and therefore, cannot be used in building mechanistic models. However, these types of datasets could be included as auxiliary resources to provide a more thorough view on disease-related mechanisms and functions.

Chapter 5

Discussion & Conclusions

High-throughput experimental techniques combined with new hypotheses regarding the character of molecular processes have been developed side-by-side with novel computational techniques to model these processes and capitalise on available datasets. In the light of this development, this thesis aimed to identify components and examine challenges of a framework for semi-automated rule-based (RB) modelling in neuroscience research. The RB approach that was placed at the centre of this thesis was selected as an advantageous method to dynamically model interactions between molecular entities on a subunit level.

5.1 Rule-based vs. ordinary equation-based modelling

The RB notation captures properties of subcellular signalling networks that are difficult to model with other frameworks such as most popular and commonly applied ordinary differential equations (ODEs). Although advantages of the RB modelling in comparison to the ODE modelling have been extensively presented and discussed in numerous reviews, no direct comparison of time courses of one molecular system encoded in the two frameworks have been presented before. Therefore, in *Chapter 2*, I directly compared the same molecular system of the immediate interaction network of dopamine- and cAMP-regulated neuronal phosphoprotein with molecular weight 32 kDa (DARPP-32) encoded in the two modelling approaches. Comparison of models was performed on two levels, model specification and time courses of selected observables.

5.1.1 Differences in model specification

The number of reactions was reduced with the rule encoding. This result was expected by virtue of the advantage of the RB language where multiple precisely defined reactions are represented by a single rule pattern. A closer look at representation of particular mechanisms revealed that this advantage is not always at play. Increase in the number of rules compared to reactions was noted for very detailed encoding of activation reactions of protein phosphatase 3/calcineurin (PP2B) and protein kinase A (PKA), termed here the "combinatorial binding" notation.

5.1.2 Comparison of time courses

The models were stochastically simulated to compare their agreement on the level of dynamics. Agreement between the models was observed regarding general characteristics of trajectories of selected observables. Direct comparison of the two model dynamics was performed by overlaying paired time courses of 15 designated observables. This revealed that for these observables that activation required the "combinatorial binding" notation (PP2B and PKA) the same reactions encoded and simulated with rules output slightly lower copy numbers than with the original ODE model. These lower abundances might be caused by the fact that activation of these observables required larger number of events in the simulation than in the original reaction notation. The number of intermediate steps between inactive and active forms of PP2B and PKA is much higher than in the ODE model. This explanation can be supported by much higher copy numbers of different half-active variants of PP2B than in its ODE equivalent observable. This observation suggests that during the simulation, activation steps "diffused" into intermediate steps. Nevertheless, additional studies are recommended to identify the exact reason of observed discrepancies between ODE and RB model dynamics, despite application of the same rate constants and kinetic law defining elementary reactions. Conclusion of this finding is that modelling the same molecular reaction system in these two formalisms might require different parametrisation of some reactions.

These differences are in fact an outcome of very detailed and explicit notation of molecular complexes and molecular species compositions in the RB modelling. This allows for exploration and observation during the model sim-

ulation as showed by employment of snapshots taken during the simulation to dissect molecular species composing the "all_Ca" observable. This transparency in the model simulation proves that the RB framework can facilitate process of model understanding. Automated specification of observables that are sums of selected time courses allows for less error-prone identification of molecular entities among molecular species and more accurate reuse of the model.

5.1.3 Modification of models

Models in the RB notation are easily modifiable as demonstrated by application of two types of model perturbations. Effects of the perturbation type that is commonly used in the ODE models, were reproduced by the RB model. For practical reasons, the second type of perturbation is only possible in the RB modelling domain. This modification involved alteration of the number of binding sites of DARPP-32 presented in two variants. In the first one, interactors of DARPP-32 bound to one site. In the second one, interactors bound to three parallel sites for each phosphorylation site. No change in dynamics was observed between the two variants of the model. This result requires further investigation. A potential modification that could change model dynamics is to significantly decrease the copy numbers of DARPP-32 compared to other interactors. An important note is that inducing this type of modification was a trivial task in the RB model. This opens a way to study effects of perturbations on a protein subunit level, dynamics of multiple binding sites with different properties, and interactions between them.

5.1.4 What kind of mechanisms to model in the RB framework?

Rules enforce explicitness of assumption about the number of simultaneously bound interactors. Exploration of more complex dependencies between individual molecular sites can be easily accomplished with different means, such as by varying their binding affinities, defining rules where binding of sites is exclusive or cooperative. Other investigations enabled by RB modelling revolve around complex formation, influence of molecular states on their function, and how local features of binding properties influence the overall system dynamics. This investigation confirmed the RB framework as suitable for dynamic modelling of signalling networks as it allows for flexible

expression of such molecular-level phenomena as distinctive functional characteristics of protein subunits, allosteric regulation, cooperative binding and cross-talks between pathways, formed by promiscuously interacting kinases and phosphatases.

In the light of these observations, two routes of investigation were undertaken in this thesis. The first one addressed the question how to analyse models that are characterised by multiplicity of molecular species generated during the simulation and determine their shift in importance due to model perturbation. The second route involved exploration of molecule-centred repositories in various biological categories that match expressibility of the Kappa language, to facilitate and accelerate process of model construction with respect to a defined biomedical inquiry.

5.2 Pipeline for analysis of RB models

The first question is addressed in *Chapter 3* by proposition of a pipeline for extended and automated analysis of RB models. The RB model of DARPP-32 network, presented in *Chapter 2*, is used to demonstrate results of the pipeline. In the pipeline procedure, observables are partitioned and scored based on sets of their time courses generated from the RB model with varied parameter sets. Selected observables are passed to global sensitivity analysis (GSA) to score impact of parameters on these observables. Scores obtained for observables and parameters are represented as a weighted network graph. The compact and unifying network representation is used to extract differences between two model phenotypes.

5.2.1 Clustering and prioritisation of observables with CorEx

Partition and scoring of observable sets were performed with CorEx, a method based on optimisation of total correlation of groups of observables assigned to hidden variables that represent clusters. Preliminary examination of CorEx was performed on repeated runs with different numbers of clusters. This demonstrated stable composition of the largest cluster regarding allocated observables. These observables are linked by the impact of the cyclic adenosine monophosphate (cAMP) signal on their abundances. These observations were made for the largest cluster of the 19-observable set. Other tested observable set with CorEx was composed of 91 automatically collected molecular species

from snapshot recordings. The largest and strongest cluster for this observable set showed complete information on composition of most dependent species driven by the cAMP signal. In both observable sets, composition of members of the largest cluster allowed to represent the same set with aggregated and generalised observable expressions. As the largest cluster was assigned with exceedingly high value of total correlation compared to other subgroups, it could be claimed that its members represent the most distinctive information content of this modelled system.

5.2.2 Observable scores for multiple time courses

Within the presented pipeline procedure, CorEx was executed on a set of time courses generated with randomised sets of parameters, as prepared to perform GSA. To aggregate CorEx-derived measures from multiple time-series, two observable scores were introduced. The first one was designed to score observables per clustering type. The second, to score observables based on all clustering measures derived from CorEx outputs. The latter scoring method was used as no distinctive clustering types emerged. The same subgroup of observables gained the top 7 scores that was earlier classified to the largest cluster among the 19-observable set.

5.2.3 Global sensitivity analysis with HSIC-based indices and network representations

These 7 observables were progressed to the next step of the pipeline where timed parameter sensitivity indices are calculated with the Hilbert-Schmidt Independence Criterion (HSIC)-based method. Integrals of area under curves defined by sensitivity scores calculated per time points served as unified sensitivity measures. Parameters and observables were combined into a weighted network with weights defined by the integrals of sensitivity measures. This representation was chosen to facilitate analysis of parameter sensitivities with respect to multiple observables. Moreover, this also enables easier representation of relations within one model and between different model conditions, e.g. induced by parameter modification. Potential of the latter was demonstrated by subtracting edge weights of the base-line model from the perturbed model to investigate the effect of the latter. Based on this operation, two types of networks were defined, a network of parameters that

gained importance and a network of parameters that lost it due to change in model conditions. The two model phenotypes were the wild-type model of DARPP-32 network and its variant with the constitutive mutation on Serine 137 (Ser137). Reordering of observable scores obtained with measures derived from CorEx demonstrated promotion of observables of DARPP-32 phosphorylated at Ser137. This result was consistent with the main effect of the perturbation where Ser137 was permanently phosphorylated. In the next step, HSIC-based sensitivity indices were calculated for the prioritised observables of the perturbed model.

5.2.4 Pipeline results agree with encoded mechanisms

Outcomes of HSIC-based sensitivity indices that reported driving parameters per observable were consistent and justifiable by the mechanisms encoded in both models. Analysis of the networks of lost and gained importance showed that a half of parameters preserved importance regardless the condition for observables representing molecular species of DARPP-32, different in each model condition. The same analysis but regarding observables present in networks of both conditions demonstrated gain of importance of parameters involved in mechanism that were first observed with the heatmap-based analysis of sensitivity scores.

5.2.5 Future perspectives

Presented in this study analysis of differences between model conditions by subtracting edge weights between two networks is a rudimentary approach that could be replaced by other more sophisticated methods from domain of differential network analysis scalable to larger networks [294, 297]. The choice of an appropriate method would depend on acquired strategy regarding decision whether to preserve or drop edges and nodes denoting observables and parameters. In this study, subsets of parameters and observables were included in the networks based on their scores. If a method for evaluation of statistical significance of parameter sensitivities and observable scores would be developed, then methods that take into account node removal and edge weights in measuring the difference could be considered in this application. In another possible setup, all parameters and observables could be included in the network and ranking of differentiated components between networks

would be performed with respect to edge weights and observable node weights defined by CorEx-derived observable scores. Calculation of sensitivity scores per each observable might be computationally expensive and in some cases even infeasible.

Although results of HSIC application were justifiable with mechanisms encoded in the model, importance of identified parameters with respect to certain observables should be subjected to further verification, in best case scenario, with experimentally-obtained evidence.

This was a first demonstration of the pipeline results that constitute a proof-of-concept and therefore, further studies with other model examples and different choices of constituent metrics would be necessary to verify and improve this approach.

5.3 Exploration of molecule-centred repositories for an ADHD-related Kappa model

The second route of investigation, undertaken in this thesis and contained in *Chapter 4*, explores molecule-centred repositories, compliant with expressivity of the Kappa language, that would facilitate and accelerate process of model construction. Contents and coverage of these datasets were studied with respect to a example of Attention Deficit Hyperactivity Disorder (ADHD), a complex neurodevelopmental disorder of relatively high prevalence and a strong genetic contribution. A list of the disorder associated genes was compiled from three resources representing different quality and types of studies. Among molecular-centred repositories, main focus was laid on protein-protein interactions (PPIs) and interactions between their subunits, DDIs. Three other resources were also employed, kinase-substrate interactions (KSIs), the BioModels database, and the REACTOME Pathway Database (REACTOME).

5.3.1 Protein and domain interactions

An in-house PPIs assembled from three databases limited to direct and experimentally obtained interactions, as reported by Molecular Interactions (MI) terms assigned to each interaction entry. A weak overlap between three leading PPI databases was observed. Moreover, spoke-expanded interactions were not marked and tagged as direct despite commonly known

uncertainty regarding directness of these interactions. A complete information regarding these interaction details could help to refine the PPI datasets with a desired level of stringency. Identifying domains in ADHD-associated proteins and linking these proteins with domain-domain interactions revealed that there is 36 times more hypothetical interactions than reported by PPIs. As existence of DDIs embedded in two proteins might not necessarily point to an actually existing interaction, obtained protein-domain interactions (PDIs) would have to be refined with other resources to eliminate conflicting domain interactions. In the study by Kim et al. [167], a structural interaction network (SIN) was defined for Yeast's PPIs based on identified 3D structures of protein complexes deposited in Protein Data Bank (PDB). The database currently stores > 130 000 macromolecular structures [405], of which around half are protein complexes [406]. Information on protein binding interfaces derived from this dataset is limited to proteins that 3D structure can be obtained therefore excludes, for instance, intrinsically disordered proteins. Prediction methods for protein binding interfaces are also based on co-crystal structures of homologous complexes (the Inferred Biomolecular Interaction Server, IBIS [407]) or domains [408, 409]. Construction of a high-quality and structurally resolved PPIs is highly desirable for RB model definition and therefore, it is an important direction in future research. Currently, as seen on an example of the INstruct database, the coverage of structurally resolved PPIs is low (6585 Human interactions) [409].

A high rate of false positive records in PPI is a common issue observed on the level of experiments [405]. A network of PPIs between ADHD-associated proteins were augmented with domain-level interaction information. Less than 40% of interactions between these proteins were potentially mediated by domains. This observation suggests that other resources for reaction details should be employed to gain greater coverage.

5.3.2 Kinase-substrate interactions

Coverage of other source of reactions was also examined with reactants of phosphorylation reactions, KSIs. Phosphorylation reactions are abundantly present in processes of cell signalling. 13.6% of all ADHD-associated genes are phophorylated by kinases that are outside of the gene list. If both kinases and substrates were among the ADHD-associated genes, only 5% of these genes

were included in the ADHD KSI network. Other resources with phosphatases could supply information on reactants of the reverse dephosphorylation reactions of identified substrates. Though PhosphoSitePlus® that KSIs were used here, does not provide similar network of phosphatases and substrates, there are other resources worth exploration and integration [388, 410].

5.3.3 Pathway gene sets

Other resources important in the dynamic modelling are pathway maps. Supported with other datasets, these were used in the past to automatically generate kinetic models encoded in the Systems Biology Markup Language (SBML) format [342, 343]. Identification of pathways where disease-related genes are overrepresented is one of essential methods of finding disease underlying processes. However, pathway databases are limited in gene-coverage and represented processes. A half of ADHD-associated genes were found in pathways of the REACTOME database. Enrichment analysis demonstrated that at most 20 of them can be found in a single pathway ("G alpha (s) signalling events"). A handful of these same genes is shared by other pathways of smaller sizes, all being subpathways of neurotransmitter release cycle and sharing the same 5 genes. This demonstrated an immense discrepancy between all explored resources, where proportion of interactions included in pathway maps is much lower than reported by high-throuput PPI datasets. These in turn appear to be much lower than the number of potential protein interactions inferred on the protein subunit level, as shown by identification of DDI among proteins.

5.3.4 Future perspectives

In the continuation of this study, identified associations between the ADHD gene set could be assembled with a rich intermediate data format to identify most information enriched subnetworks of ADHD candidate genes. Based on these modules and possibly with a pathway or connected pathways as a starting point, construction of a RB model could be attempted. However, building dynamical models requires more detailed datasets on an exact reaction network, stoichiometry of reactants and products, their copy numbers, and rate constants. Though direct measurements of rate constants for enzymatic reactions are performed on a high-throughput level [411, 412], rate constants for binding reactions (e.g. ligand-receptor, complex formation) are still technically challenging to obtain on a large scale (e.g. Surface plasma resonance,

saturation, association and dissociation binding experiments) [413, 414].

Similarly important to rate constants are time-courses recording changes of molecular species, in particular these that originate from perturbation experiments [415]. They not only allow to estimate unknown parameter values but also verify the model dynamics [416]. For instance, levels of DARPP-32 phosphorylated at Th34 and Thr75 in the Fernandez et al. [177] model, though not explicitly mentioned, were most likely estimated to match results of assays involving immunoprecipitation and immunoblotting, performed for multiple time points on neostriatial slices incubated in either dopamine (DA) [417] or glutamate (Glu) [418]. The slices were treated with various activators (forscolin) and inhibitors (e.g. roscovitin, okadaic acid, cyclosporine A) of DARPP-32 kinases and phosphatases. These not only reported changes of phosphorylated DARPP-32 but also showed what impact phosphatases and kinases have on each other and on two phosphorylation sits of DARPP-32. This allowed to infer the reaction network that underlies the model.

Immunoblotting is a common standard to measure levels of proteins in cell for multiple time points but it is a very low throughput [419]. A promising technology, similarly based on antibodies, are microwestern arrays that can quantify protein abundances for multiple time points and conditions with reduced experimental complexity and much higher throughput [420, 421]. Though mass spectrometry (MS) methods are known to be limited with respect to the number of samples [421], targeted MS based on selected reaction monitoring appears to circumvent this issue and quantifies proteins below 50 copies per cell [422]. Nevertheless, the cost, availability and labour intensiveness of such methods will not be easily circumvented to generate high-quality and high resolution time courses for large number of molecular interactions. An ideal scenario for automated model generation would be to directly and automatically build models from experimental assays and background knowledge. Such frameworks have yet to be created in the realm of dynamic and mechanistic modelling but exists for boolean modelling that require far less detailed information to construct a model. PHOsphorylation Networks for Mass Spectrometry (PHONEMeS) [423] is an example of such framework among wider research interest [424–426]. It was particularly designed to construct and train boolean logic models based on revised kinase or phosphatase-substrate interactions and phospho-MS perturbation data.

Boolean modelling is an example of a formal method that can be applied to high-throughput proteomic datasets. However, RB modelling is another such example, though much more requiring one. Even with lack of the detailed and high quality dynamic perturbation data, Stites et al. [173] constructed RB models to study recruitment of signalling proteins to epidermal growth factor receptor (EGFR) using high-throughput MS-based proteomics datasets. These datasets contained identification and quantification of protein abundances [427, 428], and equilibrium dissociation constants (K_D) derived from large-scale fluorescence polarization (FP) study of interactions between SH2 domains and ErbB receptor phosphosites [429]. The authors build 11 different HeLa cell-specific RB models by combining specific to these cell-lines data. Only site-specific interactions were included between 6 EGFR Tyrosine residues and 19 proteins containing either or both Src homology 2 (SH2) and phosphotyrosine-binding (PTB) domains. Early EGFR signalling is a well studied system and each of these models is not larger in size than the Fernandez et al. [177] model. It might be difficult to find similarly detailed information on the level of kinetic interactions for other less studied processes and cell-lines. Nevertheless, Stites et al. [173] demonstrated that though incomplete, proteomics datasets can be quickly translated and studied with a dynamic model that account for mass action kinetics and competition between sites [173]. With future increase of studies kinetically characterising interactions [429, 430] in other possibly neuronal-cell lines, the same models could be created to study disease-related mechanisms in neurobiology.

5.4 Conclusions

In the first step of this thesis, RB and ODE-based models were compared demonstrating expressiveness and opportunities lying in the former framework. A downstream pipeline was proposed to determine important observables and parameters dependent on modelled condition, that relay on two highly efficient methods for observable clustering (CorEx) and evaluation of parameter sensitivities (HSIC). Existing bioinformatics resources, suitable for development of RB models, are described and evaluated on an example of investigation of ADHD-involved mechanisms. This exploration suggested multiple obstacles relating to quality and coverage in using high-throughput datasets in detailed dynamic modelling and necessity to reach for other, more

detailed and kinetic resources to bridge the gap between mechanistic and high-throughput systems biology. Moreover, a persisting factor that limits any form of automation is lack of consensus and continuous integration of multiple resources. Solution to this can be delivered through long-term, community-based efforts towards standardisation and integration of heterogeneous datasets. Finally, I discuss future perspectives and improvements regarding proposed resources and methods, followed by example of an immediate use of RB modelling with proteomics datasets.

Appendix A

Automated translation of ODE model with Atomizer

BioNetGen (BNG) is a family member of rule-based formalisms that most closely resemble Kappa. The BNG framework is supported by an SBML-to-BNGL translation with Atomizer [176]. It is an automated translator from the Systems Biology Markup Language (SBML) model format to BioNetGen Language (BNGL). Atomizer defines molecular binding structures, implicit in reaction-based models, by relaying on molecular species naming conventions and reaction stoichiometry. This method potentially could offer an easy way to obtain a rule-based (RB) model from the original ordinary differential equation (ODE) model. The performance of Atomizer was tested on the Fernandez et al. [177] model to compare results of this automated translation to the manual translation presented in *Chapter 2*.

The automated translation was successful, though simulation of the obtained model resulted with errors, regardless application of all available options for the model simulation . The standard error output reported multiple cases of conflicting definitions and inconsistencies in naming. Notation-wise, complexity and redundancy of the resulted model encoding seemed not to be designed to be editable by human and thus, it was difficult to evaluate correctness without simulating the model. The generated model was examined with respect to definitions of agents and rules. The rule examination showed fully contextualised reaction instances (Code A.3). All combinations of dopamine- and cAMP-regulated neuronal phosphoprotein with molecular weight 32 kDa (DARPP-32) states where represented as separate species

(Code A.1, l.2 and l.3).

These results supported the necessity of the manual translation of the ODE model into the Kappa language.

Code A.1: Example of agents in BNGL generated with Atomizer

```
1 CDK5(d,d137,d34,d34_137)
2 D75(ck1,pka,pp2)
3 D(cdk5,ck1,pka)
```

Code A.2: Example of agent manually formulated in Kappa

```
1 %agent: CK1(pSite~u~p)
2 %agent: D(s, thr34~u~p, ser137~u~p, thr75~u~p)
```

Code A.3: Example of rules in BNGL for a two-step phosphorylation generated with Atomizer

```
1 von1: D(cdk5,ck1,pka)@Spine + CDK5(d,d137,d34,d34_137)@Spine
     -> CDK5(d!2,d137,d34,d34_137)@Spine.D(cdk5!2,ck1,pka)@Spine
      functionRate0()
2
3 voff1: CDK5(d!2,d137,d34,d34_137)@Spine.D(cdk5!2,ck1,pka)
     @Spine -> D(cdk5,ck1,pka)@Spine + CDK5(d,d137,d34,d34_137)
     @Spine r2_koff1
4
5 vcat1: CDK5(d!2,d137,d34,d34_137)@Spine.D(cdk5!2,ck1,pka)
     @Spine -> D75(ck1,pka,pp2)@Spine + CDK5(d,d137,d34,d34_137)
     @Spine r3_kcat1
```

Code A.4: Example of rules for a two-step phosphorylation manually encoded in Kappa

```
1 D(s, thr75~u),CDK5(a) <-> D(s!1, thr75~u),CDK5(a!1)
2                                              @'kon1','koff1'
3 D(s!1, thr75~u),CDK5(a!1) -> D(s, thr75~p),CDK5(a) @'kcat1'
```

Appendix B

Supplementary material

TABLE B.1: ADHD-associated proteins can be found in published models deposited in the BioModels database. Proteins represented with UniProtKB accession (UniProtKB AC) are listed with model names that they were found in. There are 13 proteins that association to ADHD was confirmed by at least one significant study. These are marked with asterisk.

UniProtKB Acc	Model name
P37840	Sneppen2009 - Modeling proteasome dynamics in Parkinson's disease
	Ouzounoglou2014 - Modeling of α-synuclein effects on neuronal homeostasis
	Sass2009 - Approach to an α-synuclein-based BST model of Parkinson's disease
	Morris2009 - α-Synuclein aggregation variable temperature and pH
	Cloutier2012 - Feedback motif for Parkinson's disease
	Kuznetsov2016(II) - α-syn aggregation kinetics in Parkinson's Disease
	Proctor2010 - UCHL1 Protein Aggregation
P18510*	Palmer2014 - Effect of IL-1β-Blocking therapies in T2DM - Disease Condition
	Palmer2014 - Effect of IL-1β-Blocking therapies in T2DM - Healthy Condition
P06493	Romond1999_CellCycle
	Srividhya2006_CellCycle
	Haberichter2007_cellcycle
Q13163*	Pathak2013 - MAPK activation in response to various biotic stresses
	Pathak2013 - MAPK activation in response to various abiotic stresses
P46734	Pathak2013 - MAPK activation in response to various abiotic stresses
	Mol2013 - Immune Signal Transduction in Leishmaniasis
P18754	Görlich2003_RanGTP_gradient

Continued on next page

UniProtKB Acc	Model name
P06858	McAuley2012 - Whole-body Cholesterol Metabolism
P14778	Proctor2013 - Cartilage breakdown, interventions to reduce collagen release
P32121*	Coggins2014 - CXCL12 dependent recruitment of beta arrestin
O43318	Mol2013 - Immune Signal Transduction in Leishmaniasis
Q9Y2T1	Goldbeter2008_Somite_Segmentation_Clock_Notch_Wnt_FGF
O60260*	Sass2009 - Approach to an α-synuclein-based BST model of Parkinson's disease Proctor2010 - UCHL1 Protein Aggregation
P04150	Kolodkin2013 - Nuclear receptor-mediated cortisol signalling network
P01133	Chen2009 - ErbB Signaling Schoeberl2002 - EGF MAPK Borisov2009_EGF_Insulin_Crosstalk Sivakumar2011 - EGF Receptor Signaling Pathway Jenkinson2011_EGF_MAPK Mol2013 - Immune Signal Transduction in Leishmaniasis
P01133	Capuani2015 - Binding of Cbl and Grb2 to EGFR (Early Activation Model - EAM) Birtwistle2007_ErbB_Signalling Sivakumar2011_NeuralStemCellDifferentiation_Crosstalk Bidkhori2012 - normal EGFR signalling Bidkhori2012 - EGFR signalling in NSCLC
O15055*	Weimann2004_CircadianOscillator Leloup2003_CircClock_DD_REV-ERBalpha Leloup2003_CircClock_LD Leloup2003_CircClock_LD_REV-ERBalpha Leloup2003_CircClock_DD Vasalou2010_Pacemaker_Neuron_SCN
P28562	Chen2009 - ErbB Signaling Nakakuki2010_CellFateDecision_Core Proctor2011_ProteinHomeostasis_NormalCondition Proctor2013 - Cartilage breakdown, interventions to reduce collagen release Birtwistle2007_ErbB_Signalling Bidkhori2012 - normal EGFR signalling Bidkhori2012 - EGFR signalling in NSCLC
P08913*	Thomsen1988_AdenylateCyclase_Inhibition

Continued on next page

UniProtKB Acc	Model name
	Thomsen1989_AdenylateCyclase
P45983	DallePezze2014 - Cellular senescene-induced mitochondrial dysfunction Koo2013 - Integrated shear stress induced NO production model Mol2013 - Immune Signal Transduction in Leishmaniasis Proctor2011_ProteinHomeostasis_NormalCondition Koo2013 - Shear stress induced eNOS expression - Model 3 Proctor2013 - Cartilage breakdown, interventions to reduce collagen release
O15516*	Hong2009_CircadianClock Locke2008_Circadian_Clock
P25445	Kallenberger2014 - CD95L induced apoptosis initiated by caspase-8, CD95 HeLa cells (cis/trans-cis/trans variant) Neumann2010_CD95Stimulation_NFkB_Apoptosis Kallenberger2014 - CD95L induced apoptosis initiated by caspase-8, CD95 HeLa cells (cis/trans variant) Kallenberger2014 - CD95L induced apoptosis initiated by caspase-8, wild-type HeLa cells (cis/trans variant) Kallenberger2014 - CD95L induced apoptosis initiated by caspase-8, wild-type HeLa cells (cis/trans-cis/trans variant)
P61812	Schmierer_2008_Smad_Tgfb
P49593	Li2012 Calcium mediated synaptic plasticity
P20711*	Sass2009 - Approach to an α-synuclein-based BST model of Parkinson's disease
Q9NZN5	Ung2008_EGFR_Endocytosis
P49841*	Proctor2010 - a link between GSK3 and p53 in Alzheimer's Disease Proctor2013 - Effect of Aβ immunisation in Alzheimer's disease (deterministic version) Proctor2013 - Effect of Aβ immunisation in Alzheimer's disease (stochastic version) Sivakumar2011_WntSignalingPathway Kim2007 - Crosstalk between Wnt and ERK pathways Goldbeter2008_Somite_Segmentation_Clock_Notch_Wnt_FGF Sivakumar2011_NeuralStemCellDifferentiation_Crosstalk
O14974	Maeda2006_MyosinPhosphorylation
P19367	Sengupta2015 - Knowledge base model of human energy pool network (HEP-Net)

Continued on next page

UniProtKB Acc	Model name
P07332	Sengupta2015 - Knowledge base model of human energy pool network (HEP-Net)
P07101*	Sass2009 - Approach to an α-synuclein-based BST model of Parkinson's disease
P53597	Sengupta2015 - Knowledge base model of human energy pool network (HEP-Net)
P21397*	Sass2009 - Approach to an α-synuclein-based BST model of Parkinson's disease
O15534	Leloup2003_CircClock_DD_REV-ERBalpha Leloup2003_CircClock_LD Leloup2003_CircClock_LD_REV-ERBalpha Leloup2003_CircClock_DD Vasalou2010_Pacemaker_Neuron_SCN
P06733	Sengupta2015 - Knowledge base model of human energy pool network (HEP-Net)
P01100	Nakakuki2010_CellFateDecision_Core Mol2013 - Immune Signal Transduction in Leishmaniasis Swat2004_Mammalian_G1_S_Transition Proctor2013 - Cartilage breakdown, interventions to reduce collagen release
P18089*	Thomsen1988_AdenylateCyclase_Inhibition Thomsen1989_AdenylateCyclase
O14727	Albeck2008_extrinsic_apoptosis Legewie2006_apoptosis_NC Legewie2006_apoptosis_WT Rehm2006_Caspase
Q9Y6D9	Ibrahim2008_MCC_assembly_model_KDM Ibrahim2008_Cdc20_Sequestring_Template_Model Ibrahim2008 - Mitotic Spindle Assembly Checkpoint - Convey variant Ibrahim2008 - Mitotic Spindle Assembly Checkpoint - Dissociation variant
Q9UD71	Li2012 Calcium mediated synaptic plasticity Fernandez2006_ModelA Fernandez2006_ModelB
Q05940	Sass2009 - Approach to an α-synuclein-based BST model of Parkinson's disease
P18825*	Thomsen1988_AdenylateCyclase_Inhibition Thomsen1989_AdenylateCyclase

Appendix C

Building models for biopathway dynamics using intrinsic dimensionality analysis

Authorship contribution

I chose a biological context of this study based on RB models of Suderman and Deeds [123]. I simulated the models and prepared time courses for further analysis, wrote a great part of this report with an exception of CorEx description and fragments of choice rationales that was written by S. Garg and fragments of "Introduction" and "Chaos Time Series Analysis" by V. Dzutsev. S.Garg proposed CorEx as a method for dimensionality reduction and run it with provided time courses. V. Dzutsev performed nonlinear time series analysis. L. Condon proof-read the report.

Building models for biopathway dynamics using intrinsic dimensionality analysis

Emilia M. Wysocka[1], Valery Dzutsati[2], Tirthankar Bandyopadhyay[3], Laura Condon[4], and Sahil Garg[5]

[1]University of Edinburgh, UK
[2]Arizona State University, USA
[3]CSIRO, Australia
[4]Colorado School of Mines, USA
[5]University of Southern California, USA

September 25, 2015

COMPLEX SYSTEMS
SUMMER SCHOOL 2015
SANTA FE INSTITUTE
NM USA

Contents

1 Introduction and project motivations

Extensive development of technologies and methods related to data acquisition, sharing and storage have made analysis and knowledge discovery unprecedentedly challenging. For instance, big data has become increasingly common in social sciences and requires new techniques of analysis, including non linear time series approaches. One of such recent examples of a challenging dataset is the data on rebel violence in the volatile Russian North Caucasus region [26]. The dataset has recordings of incidents of rebel violence on weekly basis for every town and village of the region. Overall, this resulted in over 1 million observations with nearly 200 variables, approximately 200 million data points.

An important task for many if not all the scientific domains is efficient knowledge integration, testing and codification. It is often solved with model construction in a controllable computational environment. In spite of that, the throughput of *in-silico* simulation-based observations become similarly intractable for thorough analysis. This is especially the case in molecular biology, which served as a subject for this study.

In this project, we aimed to test some approaches developed to deal with the curse of dimensionality. Among these we found dimension reduction techniques especially appealing. They can be used to identify irrelevant variability and help to understand critical processes underlying high-dimensional datasets. Additionally, we subjected our data-sets to nonlinear time series analysis, as those are well established methods for results comparison.

To investigate the usefulness of dimension reduction methods, we decided to base our study on a concrete sample set. The example was taken from the domain of systems biology concerning dynamic evolution of subcellular signalling. Particularly, the dataset relates to the yeast pheromone pathway and is studied *in-silico* with a stochastic model. The model reconstructs signal propagation stimulated by a mating pheromone.

In the following sections we will elaborate on the reason of multidimensional analysis problem in the context of molecular signalling. Next, we will introduce the model of choice, simulation details and obtained time series dynamics. A description of used methods followed by a discussion of results and their biological interpretation will finalise this report. This study is a preliminary analysis of the dataset, future work will expand on the results presented here.

2 Working with example

2.1 Problem view: combinatorial explosion of cell signalling systems

As with all signal processing systems, cell signalling is characterised by signal related functionalities, such as input fidelity, output specificity, signal amplification, the sensitivity and diversity of response or the flexibility of

reaction [1]. These highly sophisticated functions produce complex systems embodied by the combinatorial explosion of molecular interactions and states [12, 2].

On the lower level, cell signalling depends on formation and interactions of multi-subunit complexes, mainly formed by interacting proteins. They are composed from often numerous and autonomously folding blocks called domains, acting as protein functional interfaces. Importantly, protein activity is determined by multiple post-translational modification sites (phosphorylation, acetylation, ubiquitination), transitionally changing their states. For example, lets consider an ubiquitously present Epidermal Growth Factor Receptor (EGFR), which has 9 sites resulting in 512 possible states ($2^9 = 512$, on- and off-state). Furthermore, each site has at least one binding partner rising the value of single receptor protein states to 19,683 possibilities ($3^9 = 19,683$). The large number of possible states, even within this relatively simple system is one of the key challenges for mechanistic modelling of signalling networks. Traditional equation-based models are capable of representing only extensively studied and limited size signalling circuits. Any larger integrative models become intractable, impossible to reuse or even proofread [19]. These problems have been addressed by rule-based modelling methods embodied by flexible languages such as Kappa [3] and BioNetGen [6], facilitating the creation of large and complex dynamical models. In contrast to the other modelling techniques, in rule-based models the system emerges with time, often showing unpredictable behaviour arising from elementary reaction rules. However, their construction and analysis often limit their potential application. For instance, even though provided with visualization tools for static and causal analysis, a modeller has to resort to a self-assembled battery of tests trying to unfold the complexity of results [24].

2.2 Yeast pheromone response pathway model

In the domain of molecular biology the yeast pheromone cell cycle is an extensively studied example, both *in-vivo* and as a computational model. It's often used to test hypotheses and investigate details related to mechanisms of signalling processes, such as dynamical pathway adaptation to demanding environmental conditions [20], evolutionary preserved functional units (G-protein coupled receptor signalling [4], mitogene-activated protein kinase [17]), signal-noise decoupling [4] and information transmission [29].

Saccharomyces cerevisiae yeast, is a model species, capable of sexually reproducing in pairs of opposite sexes (type a and α). The mating signal is communicated by either of the cell type through pheromone release (*a-factor*) [20]. The model used in this study relates to a subcellular signalling activated in the other cell in response to the stimulus [24].

The pathway represents canonical mechanisms of the subcelluar signal propagation, such as G-protein activation via a GPCR, which is stimulated by pheromone ligands. The scaffold protein (Ste5) is recruited to the cell surface.

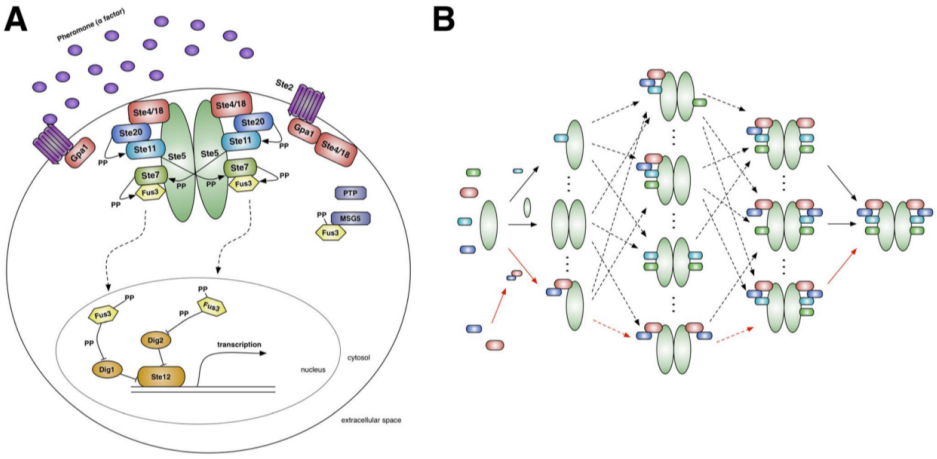


Figure 1: A: Usual scheme of hierarchically structured molecular machines B: Potential combinations of complexes appearing over the simulation. The red-arrow path represents the possible way of construction of decamer complex. Source: [24]

Its major role is to insulate the kinase phosphorylation cascade from activating other related pathways. Ste5 dimerizes and aggregates five more proteins that phosphorylates each other forming an activation cascade. The last one is doubly activated mitogene-activated protein kinase (MAPK, Fus3) that travels to the nucleus and releases the transcription factor (TF) from its inhibitors. In this way TF transcribes genes regulating yeast mating behaviour.

The study is examining the established hypothesis that signals in cells are propagated via well defined complexes of molecular machines rather than loosely assembled and polymorphic ensembles.

As it was shown, even though a conserved structure of decameric complex was hardly present in the ensemble model over repeated simulations, the signal was uninterrupted leading to St4 synthesis. Furthermore, contrary to the machine model, the ensemble model was able to replicate the experimental observation of combinatorial inhibition of phosphorylated Fuss3 (Fus3pp), when a copy-number of St5 was increased 60 fold. Models were built with the rule-based formalism that allows us to sample the sets of possible protein complexes the model can produce, without explicitly imposing the set of species that are formed [24]. More details about the formalism are in the next section. The code with the models' implementation is in the public domain and can be found as one of the attached files to this paper.

2.3 Rule-based modelling

The subject of signalling pathways and networks has already been addressed by many modelling formalisms. However, one significant advantage of the

rule-based (RB) modelling is that it is able to express an infinite number of reactions with a small and finite number of rules, i.e. a single reaction rule and its parameters generalize a class of multiple interactions. In all of the other modelling methods every chemical species has to be specified in advance which is highly problematic for species with dozens of phosphorylation sites and many possible states. This limiting factor makes these methods inappropriate for modelling large-scale complex dynamical systems.

RB modelling is a method for the formal representation of combinatorially complex signalling systems in both a qualitative and quantitative way. The major idea is to replace equations with interaction rules. A rule representation is a graph-rewriting, where a graph specified on the left-hand-side is a pattern to be matched to instances in the current "mixture" of graphs and transformed into graphs specified on the right-hand-side. Matching should satisfy *embedding*, i.e. injections on agents (graphs) with the preservation of names, sites, internal states and edges [3]. In the rule-based language nomenclature, "agents" are most elementary molecules and "species" are agent complexes having particular states. A model can be translated into a system of ODE equations or simulated with a stochastic algorithm. In the latter case, system trajectories are created by rule selection at each time step, which is applied probabilistically, based on reaction rates and the initial/current copy number of agents [12]. Immediate consequences of this formulation are different levels of rule contextualisation ("don't care don't write"), without obligation of *ad hoc* assumptions about the system, modularity, reusability and extensibility of the modelling process [19]. Furthermore, the ability to capture a protein as a graph with (binding) sites (e.g. domains) that have internal state(s) (e.g. phosphorylated) gives a sufficiently expressive system to capture all of the principal mechanisms of signalling processes (e.g. dissociation, synthesis, degradation, binding, complex formation [18]) as well as insight into site-specific details of molecular interactions such as affinities, dynamics of post-translational modifications, domain availability, competitive binding, causality and the intrinsic structure of interactions.

RB modelling originates from concurrency system representations and as such has the ability to capture dependencies, causality and conflicts in biological interactions (overlooked by concentration-based ODEs). In other words, precedences occurring along trajectories (stories) reveal competing events leading to a final state [3]. In the Kappa language, this feature is supported by the syntax for graphical analysis provided in the simulation tool. Among these are diagrams with causal flows, flux and influence maps as well as contact maps [Figure 2] that facilitate the process of modelling. The causal flow diagram shows dependencies and conflicts in tracking indicated species and the flux maps [Figure 13], negative/positive activity transfers between rules with the quantitative contributions on edge weights, both generated on the fly during a simulation [7]. At any time of a simulation, a snapshot can be taken to record the collection of species existing at that time.

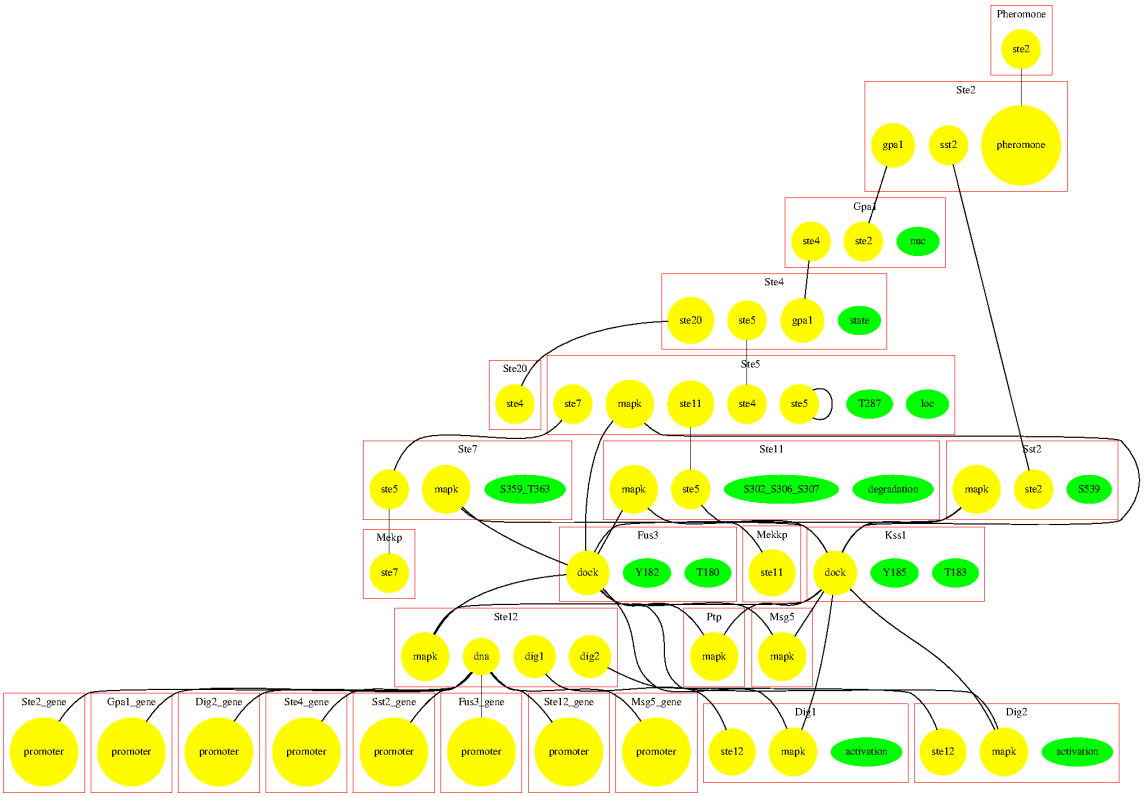


Figure 2: Contact map defined without running the simulation with KaSa software accompanying KaSim4.0 simulation tool. Yellow circles denote agents sites, green circles agent states, and edges all potential connections between species.

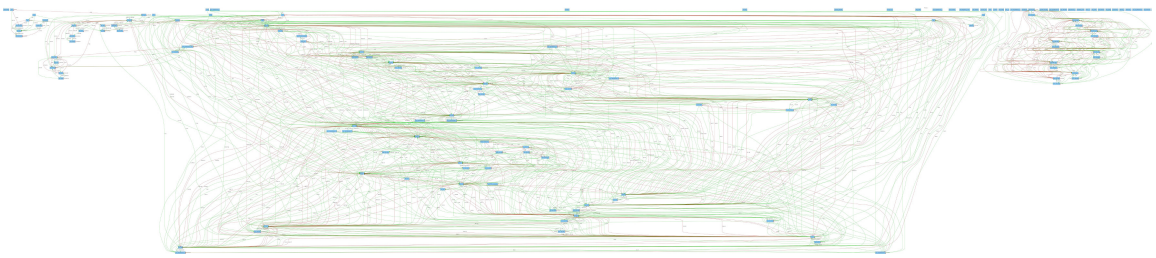


Figure 3: Flux map for pheromone pathway model in steady state simulated with KaSim4.0

2.4 Datasets and simulations

Our time series datasets report changes of indicated molecular species' copy-number over 13,800 time points. Stochastic simulations were run for 4,600 sec with 3 time points recorded per second. The system was first simulated over 1,000 sec to reach a steady state and the initial mixture of protein complexes. Afterwords, a pheromone stimulus was introduced and the system was simulated for another 3,600 sec.

Variables in the rule-based syntax are called "observables" (`%obs:`) and are specified in a separate code block. A single observable can be mapped to one or more rules conditioned on the level of its particularity. Hence, all the types of created species not indicated as observables, become intractable. For instance, the observable `%obs: Fus3PPFus3(T180~p,Y182~p)`, which is a double phosphorylated MAPK kinase Fus3, is associated with 14 rules of the following form:

- ```
  Fus3(dock!1,T180~p,Y182~p),Sst2(S539,mapk!1)
  -> Fus3(dock,T180~p,Y182~p),Sst2(S539,mapk)
  ```
- ```
  Ste7(ste5!2,S359_T363~pp,mapk!1),Ste5(ste7!2),
  Fus3(dock!1,T180~p,Y182~p)
  -> Ste7(ste5,S359_T363~pp,mapk),Ste5(ste7),
  Fus3(dock,T180~p,Y182~p)
  ```
- ```
  Fus3(dock!1,T180~p,Y182~p),Ste11(mapk!1)
  -> Fus3(dock,T180~p,Y182~p),Ste11(mapk)
  ```
- ```
  Ste5(ste7!1),Ste7(mapk!2,ste5!1,S359_T363~pp),
  Fus3(dock!2,T180~p,Y182~u)
  -> Ste5(ste7!1),Ste7(mapk!2,ste5!1,S359_T363~pp),
  Fus3(dock!2,T180~p,Y182~p)
  ```
- `...etc.`

However, as it is with the model specification, as it is infeasible to observe all potentially important variations of species, we have to resort to what we know we want to observe.

Therefore, the considered dataset consists of standard 31 variables, patterned after the original paper. There is also an extended 977 variable set but it has yet to be explored with parallel computations. This number was dictated by the snapshot of all existing species at the pick of the simulation (~1,000 sec after the stimulus appearance) used then as a list of observables in the simulation.

2.4.1 Perturbed model

To compare the outcome of applied methods, the model was simulated in two states, which are called here "perturbed" and "unperturbed". By the unperturbed

model we call the standard "wild type" pathway dynamics. The perturbed one relates to an experimentally observed phenomenon of combinatorial inhibition. It occurs when the copy-number of protein scaffold is largely increased and impossible to fully assemble the complex that doubly activates Fus3 because all available members of the complex are used up on too many scaffold proteins.

2.4.2 Simulator

Models written in Kappa language are supported by KaSim simulator. By default, reaction rates are computed applying the law of mass action [2] but can easily be adjusted to follow any kinetic law (e.g. Michaelis-Menten, Hill's Law). What can be found under the hood is a direct particle-based variant of Gillespie's method. A general version of Gillespie's method, also called exact stochastic simulation algorithm (SSA) or kinetic Monte Carlo is a common simulation method for modelling time-evolution of stochastic chemical reaction systems. Numerical stochastic simulations are known to be computationally intensive and a lot of efforts have been made to improve their efficiency [10]. The most popular and effective solution, implemented in KaSim, is called "network-free" because rules transforming reactant into products are applied directly at runtime to advance the state of a system. As a result, it does not have to generate the full reaction network beforehand and is therefore independent of its size [13].

3 Applied methods and results

3.1 Correlation Explanation

3.1.1 Choice rationales

Since a rule representation may vary in generalization, it can be applied to more then one reaction that satisfies it. In other words rules serve as the reaction and species generator. It results in the unpredictability of species types and their importance emerging over time. Especially, if the model is of a large magnitude.

On the other hand, the intrinsic modularity of Kappa syntax opens the path to large integrative models, gradually assembled from the collections of reusable rule-based syntactic modules.

However, models are currently built in a fully controllable and stringent fashion. It leaves the notion of modularity and its experimental aspect risky and unexplored. Thinking ahead, the rule notation can be understood as an updatable, machine readable and executable knowledge representation and storage, replacing the usually manual revision of papers required in the model construction [15]. We could allow for uncontrollable, collective model growth in a form of rule stacks and then verify inner links and hierarchies in the system. That could guide an automatic trimming of the model size. Hence, the question is whether and how we could restrict a model to only these rules which are most informative.

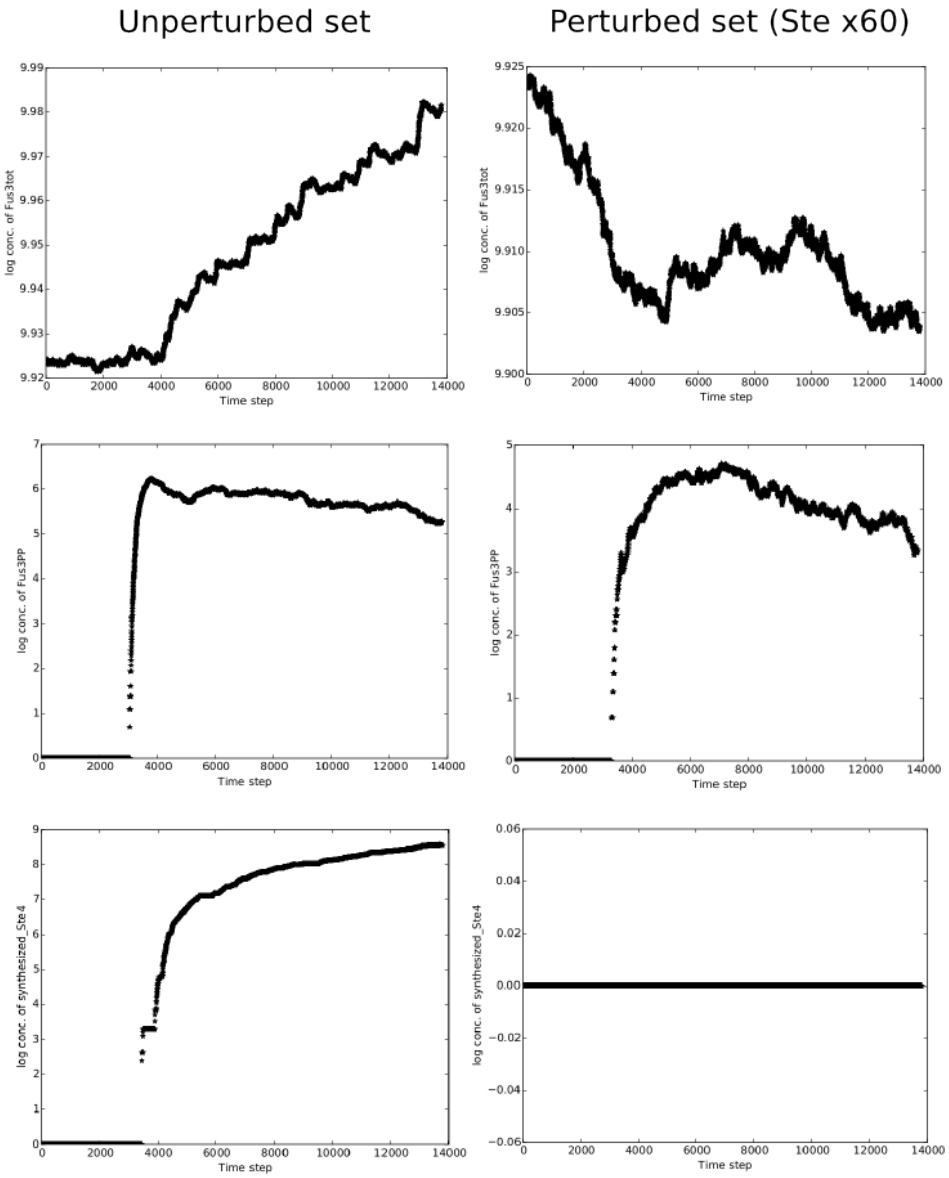


Figure 4: Model dynamics in unperturbed and perturbed states for characteristic protein species. The perturbed ensemble model showed a decrease in Fus3 activation (Fus3PP) being a key observation of the combinatorial inhibition. As we can see, the synthesis of St4, which happens in the nucleus, was inhibited under perturbed state (plot with a flat line).

Likewise the question lies what exactly does it mean to say a species is "important" or "informative" and what do groups of biologically important species share with each other? Are they strongly interlinked modules of the system? Could they guide the rule-based modularity idea? The last question is especially important, since the scope of elementary parts of RB model is not yet clear. Are these three, four, five reaction rules? Is there any other measure of mechanisms granularity?

Facing these kind of questions, we opted to test one of recently realised methods Correlation Explanations (CorEx) that applies information theoretic objective to learn a hierarchy of more abstract representations of the data.

Having the hierarchy of latent variables, we can pose more precise questions, such as:

- What subset of species has common underlying dynamics?
- How strong is the correlation between species grouped under the same latent variable?
- How many underlying hidden causes can be identified out of the observed high-dimensional species dynamics?

and finally:

- What could be the meaning of these "hidden causes" for molecular signalling?

The algorithm was previously applied in a biological context for identification of targets for a cancer therapy [23]. Furthermore, a similar method mentioned in CorEx paper, called the information bottleneck, was previously applied for trimming of gene ontology (GO) [14] to compress the data into a smaller representation. In contrast, in CorEx the redundant information is preserved ignoring uncorrelated random variables [27].

3.1.2 Method description

In this section, we discuss an information theoretic approach for building a model on dynamics of the species concentrations. This method, proposed recently for a general domain [27, 28], learns a hierarchy of latent variables that maximally inform correlation between the observed species dynamics. Herein, correlation refers to mutual information between a set of variables, and not just a linear correlation.

Before we delve into the details of the method for our specific settings, it should be noted that we disregard the time series nature of species copy-number dynamics in this method application.

Let G be a set of random variables representing copy-numbers for all the species. Then, X_G is a joint random variable on G. For a species i, all the

copy-number values of the time series are assumed to be independent samples of a random variable X_i. As such, we can see that there is a contradiction since consecutive samples in the time series would have a correlation (not i.i.d.). For obtaining uncorrelated samples, one can take sub-samples of the time series, either at uniform interval or using any other relevant technique.

Following the notations in [27], total correlation $TC(.)$ between a set of variables X_G is defined as below.

$$TC(X_G) = \sum_{i \in G} H(X_i) - H(X_G) \tag{1}$$

$$TC(X_G) = I(X_1; \cdots ; X_g) \tag{2}$$

Here $H(X_i)$ is entropy on a random variable X_i; and $H(X_G)$ is a joint entropy on X_G. Another interpretation of $TC(X_G)$ is that it is mutual information, $I(.)$, between all the variables in the set G. Typically mutual information is expressed between a pair where each element of the pair can be a set of random variables. Here, we are instead expressing mutual information between a g dimensional triplet of random variables, where g is a number of random variables in the set G.

In our problem of learning a model of species dynamics, evaluating mutual information (or total correlation) between all random variables would not be of much value. We are instead interested in evaluating mutual information between some subsets of species. However there are two problems along these lines: i) we do not know for which subsets of species we should evaluate mutual information and there can be a large number of permutations to explore (depending on the size of a subset and the G set); ii) non-parametric estimation of mutual information between random variables is a hard problem [16, 25, 22, 9]. To tackle these, we formulate our algorithm such that; i) we assume the individual species copy-number variables X_i to be Gaussian; however, we do not assume that *the set of variables* has to be Gaussian (the later is a stronger assumption); ii) we are interested in only those subsets where variables have low mutual information conditioning on a latent variable (or high mutual information between variables explained by a latent variable).

Along these lines, let us define a new information theoretic quantity $TC(X_G; Y_F)$.

$$TC(X_G; Y_F) = TC(X_G) - TC(X_G|Y_F) \tag{3}$$

$TC(X_G; Y_F)$ is a total correlation (or mutual information) in the set of random variables X_G explained by a set of latent variables Y_F. $TC(X_G|Y_F)$ is a total correlation between the random variables X_G that can not be explained by Y_F, i.e. conditional total correlation (conditional mutual information). If the latent variables Y_F can explain the total correlation in X_G perfectly, then $TC(X_G|Y_F) = 0$. Ideally, we would like to learn Y_F if exists. Thus intuitively, optimal Y would

correspond to minimum of $TC(X_G|Y_F)$. In our formulation, we can express optimization of Y_F as optimizing conditional distributions $P_{Y|X}$.

Let us first consider optimization of a single latent variable Y, and then generalize it later.

$$\arg \max_{Y:p(y|x_G)} TC(X_G;Y) \; s.t. \; |Y| = k \tag{4}$$

Here Y is a discrete random variable; x_G is a sample of the random variable X_G and y is sample of Y. We optimize Y by learning the conditional distribution $p(y|x_G)$. Now, we further extend it for multiple latent variables, where each latent variable explains total correlation in a subset of the species concentration variables.

$$\arg \max_{G_j, p(y_j|x_{G_j})} \sum_{j=1}^{m} TC(X_{G_j};Y_j) \tag{5}$$

We have introduced m latent variables here with Y_j explaining correlation between random variables in the corresponding subset $G_j \subset G$. Here these subsets can have an overlap.

Optimizing the above objective function seems hard. However, as explained in detail in [27, 28], it can be solved very efficiently for practical purposes. We omit these optimization details and refer readers to the original papers introducing this algorithm for the first time [27, 28]. Computational complexity of the method is linear with respect to the number of samples and number of variables in the set G. Furthermore, as an unsupervised method, it requires no assumption about the learned model. The code implementation for this algorithm is publicly available from the original authors [1].

3.1.3 Results

The CorEx algorithm was applied both to perturbed and unperturbed datasets and yielded two results with a single layer of hidden variables. In both cases, presented results were the maximal values the data sets could be divided to. Further increase of the number of hidden units did not change their values.

For both sets [Figures 6 & 5] CorEx found 6 latent variables, where 8-9 out of 31 biologically plausible variables were expected.Biologically plausible variables were thought to be all these observables that contained Ste5 scaffold protein, known to be a nucleation point of the system [24]. However, the preliminary intuitions did not align perfectly with the algorithm results.

In the unperturbed data tree we can distinguish two important groups, ‘0’ and ‘1’. They are recognisable in the perturbed data tree as they preserved half of their members from the former set. However, contrary to the unperturbed data tree,

[1] https://github.com/gregversteeg/CorEx

where the members of both groups have similarly balanced strong relations, the perturbed set shows far uncertain correlations, mostly concentrated in the group '0'.

3.1.4 Interpretation and analysis

The interpretation of the results was conducted on two levels. The first one is based on the biological knowledge about the process. The second one is supported by the dynamic analytical tools provided by the KaSim simulator.

Generally speaking, the CorEx algorithm successively subsets data into a defined number of latent variables guided by species dynamics. Results appears to be consistent with the differences between perturbed [Figures 9 & 10] and unperturbed models [Figures 7 & 8]. The group ordering, referring to the strength of inter-correlations, shows which event takes the lead in two cases. Earliest events upstream to the formation of signalling cascade appeared to be the leading ones in the perturbed simulation. This is consistent with the fact that phosphorylation of Fus3 kinase distinctively drops when the amount of Ste5 protein scaffolds competing for binding kinases increases [Figure 4 in Section 2.4.2]. As the Fus3 phosphorylation was not entirely blocked, the second latent variable relates to events leading to Fus3 phosphorylation. Thus it is more consistent with the group '0' apart from transcription in the nucleus, which was inhibited in the perturbed data.

Owing to the static causal analysis provided by the simulation software, we can ask whether important observables relate to frequently executed rules. The most powerful visualization output is a flux map, which tracks the overall influence of rule applications on each other [7]. It is a directed and weighted graph with rules as vertices and edges annotated with positive or negative weights [Figure 13]. Dependent on simulation parameters (selected time or number of events), a flux map might vary in structure (for details of our simulation parameters see section 2.4).

Both untrimmed graphs for unperturbed and perturbed models had 233 vertices but they differ from each other in the edge number (unperturbed- 2,753, perturbed- 2,422). Weights range from 0 to 407,172,203. An important note is that vertices are rules. Hence, to compare them with the output of CorEx, thus subsets of observables, first observables had to be mapped to rules they referred to [Figure 11 & 12]. The weight cutoff varies with inverse proportion to the number of observables in flux map subgraphs. Therefore, we compared different subgraphs by gradually removing less and less vertices given a set of thresholds for weight values. The aggregated results are presented in the Figure 13. As we can observe, the frequency of rule application relates to subsets obtained with the CorEx algorithm but cannot explain them fully.

We stated some questions in the Section 3.1.1, which we would like to comment on or even answer to in the following part. We have learned that the algorithm used on time series datasets divided the species into most important ones

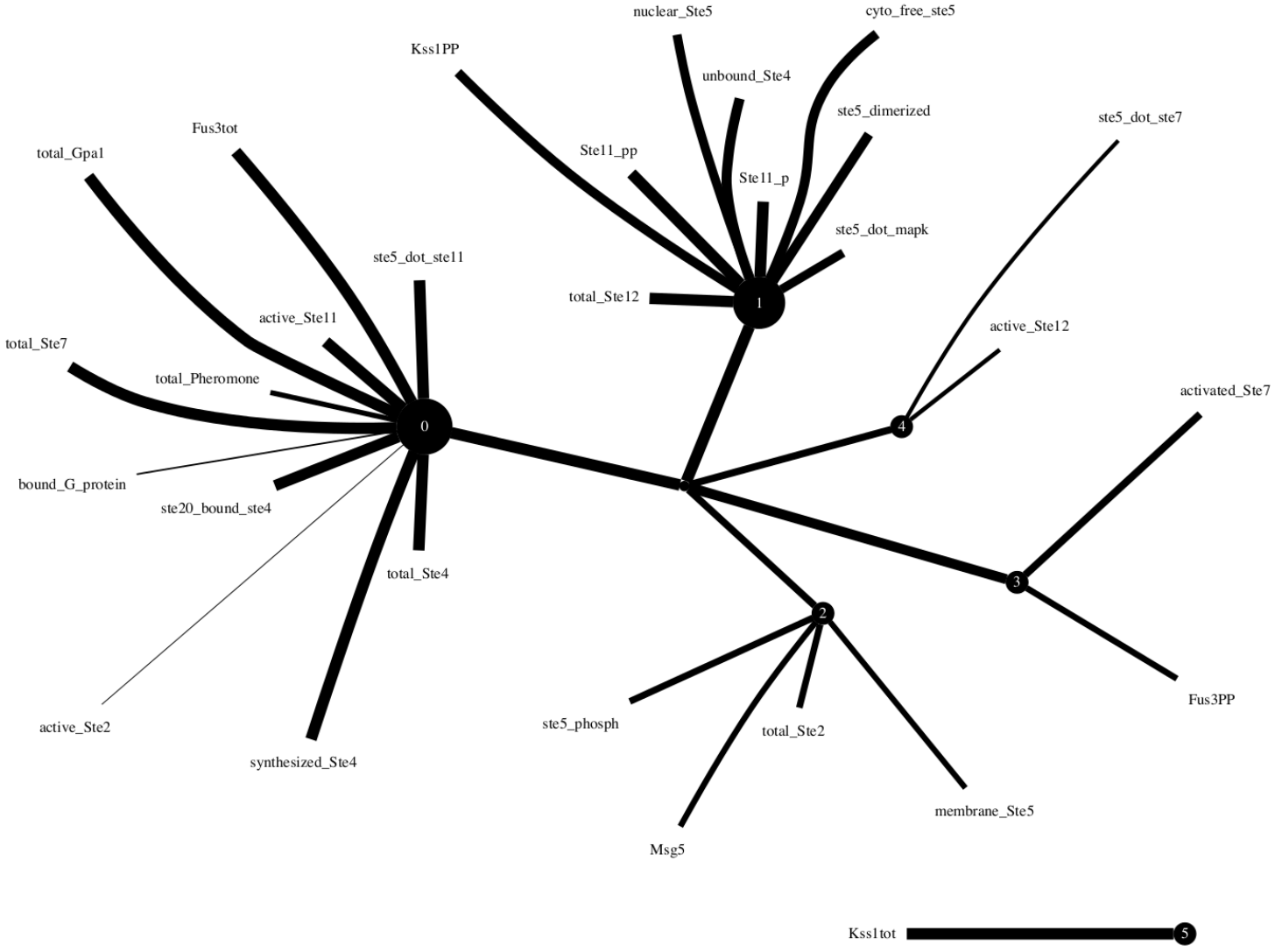


Figure 5: Result of 31-variable dataset **without** perturbation. Intrinsic dimensionality was found to be 6. Variable numbers are shown in the middle node of each group. Edge weights leading from a group centre to its member are dictated by its explanatory contribution to remaining group members

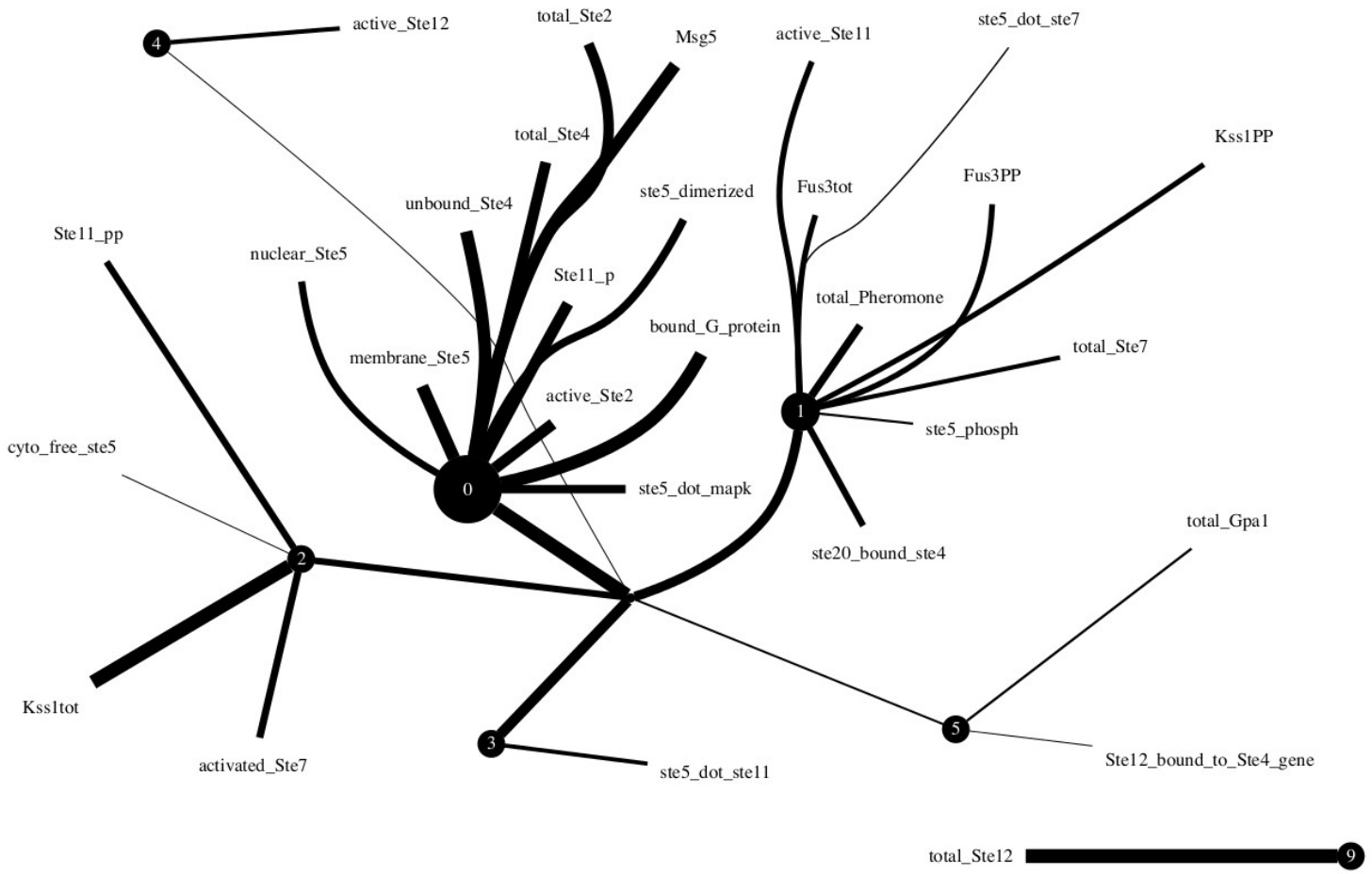


Figure 6: Result of 31-variable dataset **with** perturbation. Intrinsic dimensionality was found to be 6. Variable numbers are shown in the middle node of each group. Edge weights leading from a group centre to its member are dictated by its explanatory contribution to remaining group members

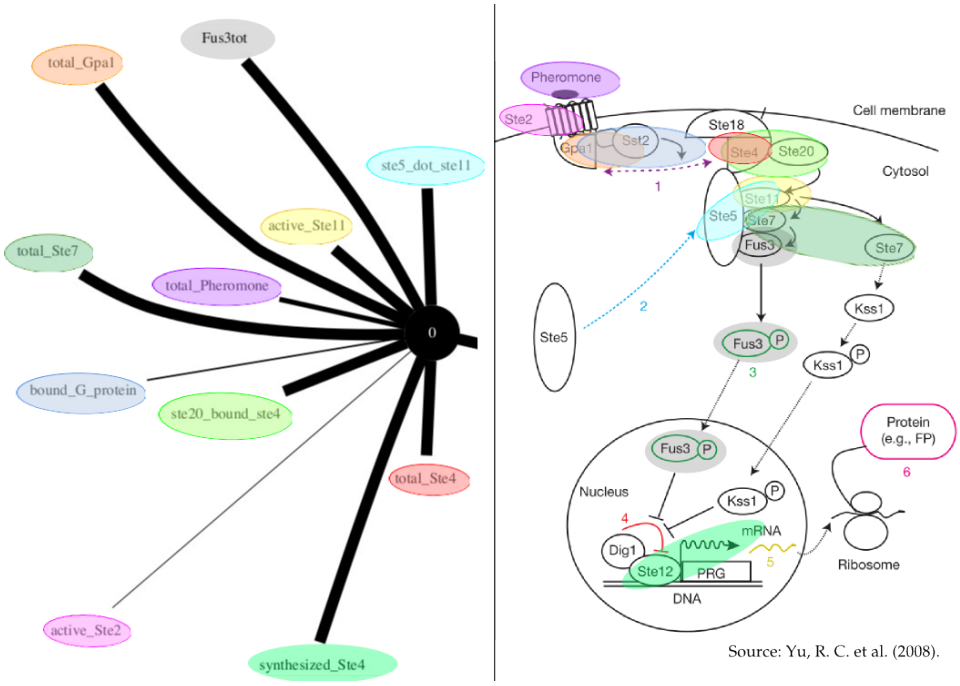


Figure 7: On the left, a fragment of unperturbed data-tree with Group '0'. On the right, the process diagram for a comparison. A biological interpretation of the strongest group '0' refers to the most important steps indicating critical events in the successful signal propagation. As the authors argued, the assembly of decamer involving Ste5 dimerization does not belong to the most crucial events guaranteeing the signal transfer.

over the entire course of time series, with results depended from the outcome of signalling process. Hence, it did not inform us about intrinsic modularity of the system, what would relate to more "horizontal" division of time courses (when looking at the process diagram). Perhaps the considered system is far too small thus interlinked to observe invariable modules among species (encapsulations). Hence, the result might be then more correctly named as a form of "compression". Furthermore, given the limited number of experiments and the model size and its character, we are not yet ready to precisely answer the question of biological meaning related to the importance and informativeness indicated by the algorithm.

3.2 Chaos Time Series Analysis

To compare results with the outcome of CorEx algorithm and discover other aspects hidden in our data, we applied methods of nonlinear time series analysis [11]. Similarly, we used both the perturbed and unperturbed datasets (for more details about used datasets see Section 2.4). To bypass an obvious division into a pre- and post-pheromonal stimulation, we cut the beginning 1,000 sec and used

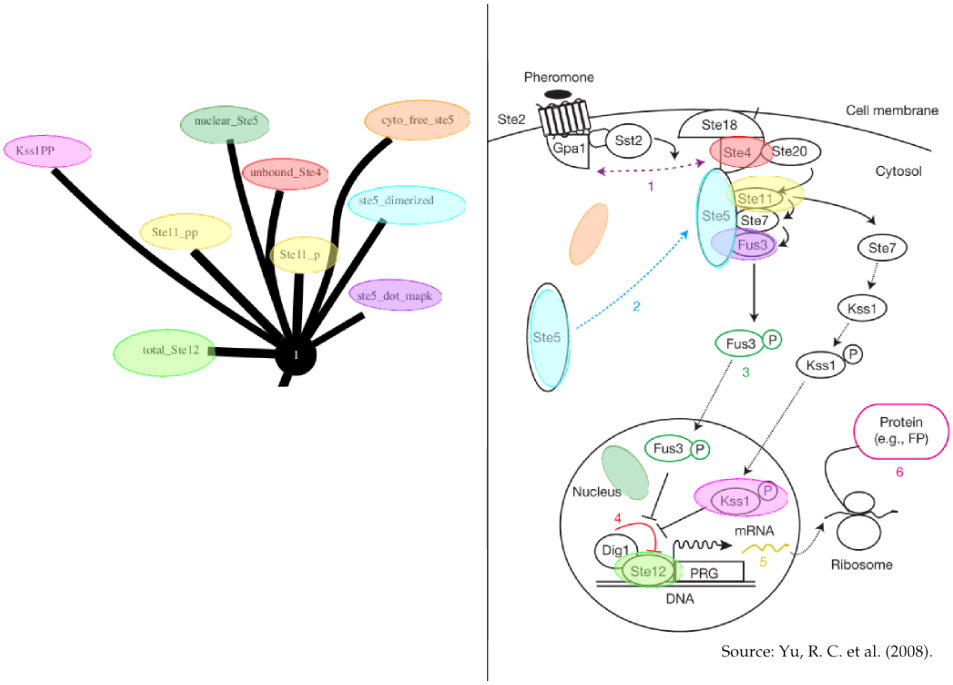


Figure 8: On the left, a fragment of unperturbed data-tree with Group '1'. On the right, the process diagram for a comparison. The second highly scored group indicates less vital events, related to dimerization, and the impact of Kss1 kinase on the Ste4 activation.

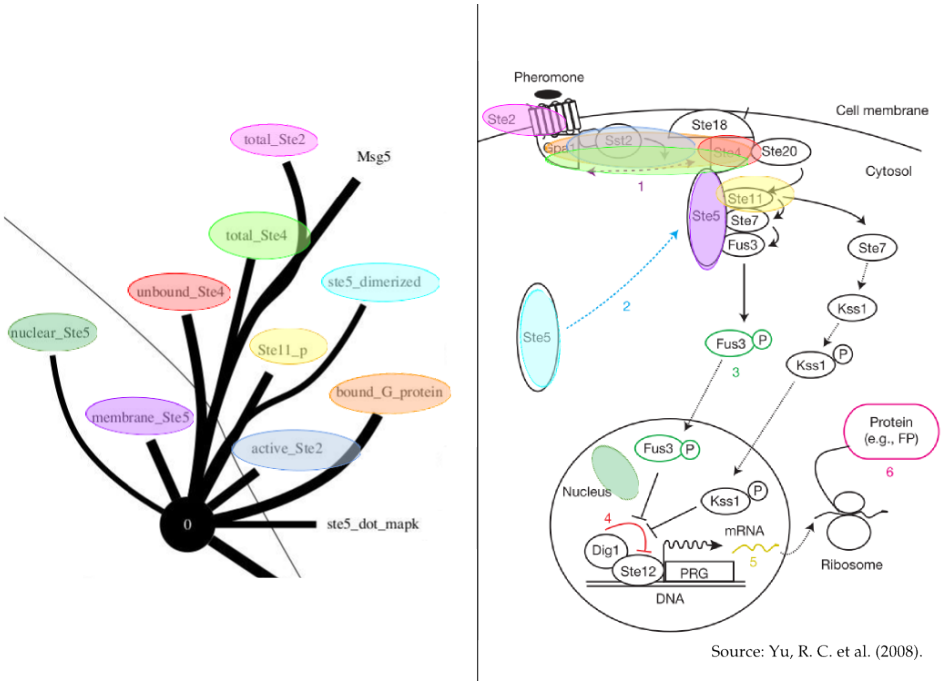


Figure 9: On the left, a fragment of perturbed data-tree with Group '0'. On the right, the process diagram for a comparison. The strongest group indicates the earliest events located upstream to the Fus3 poshorylation (observable called Fus3PP), preceding the complexation step

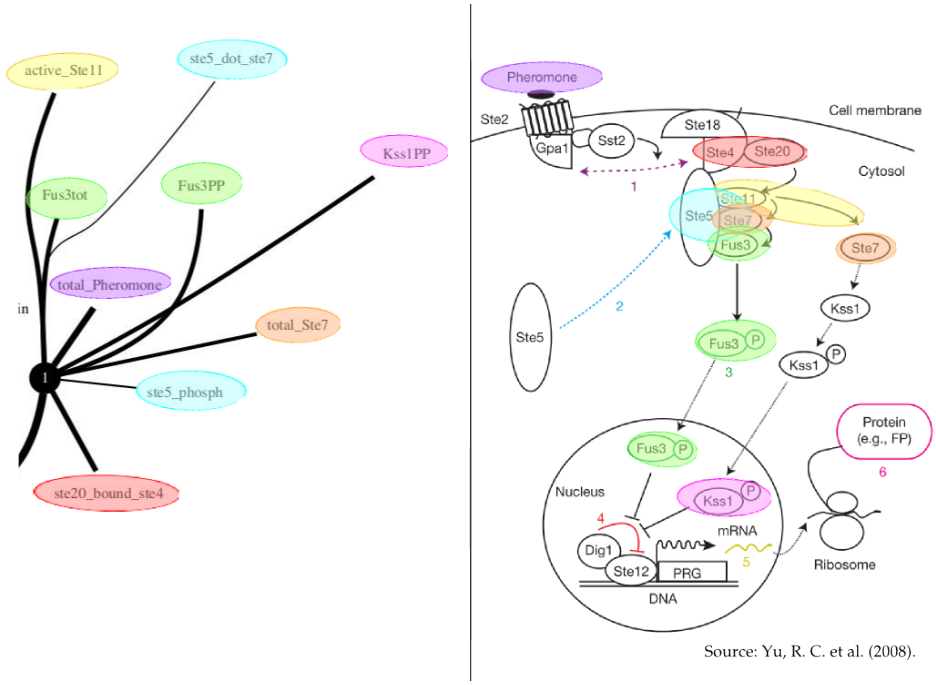


Figure 10: On the left, a fragment of perturbed data-tree with Group '1'. On the right, the process diagram for a comparison. The second strongest group of perturbed data tree reflects weak correlation between members and the unsuccessful activation of transcription factor St4.

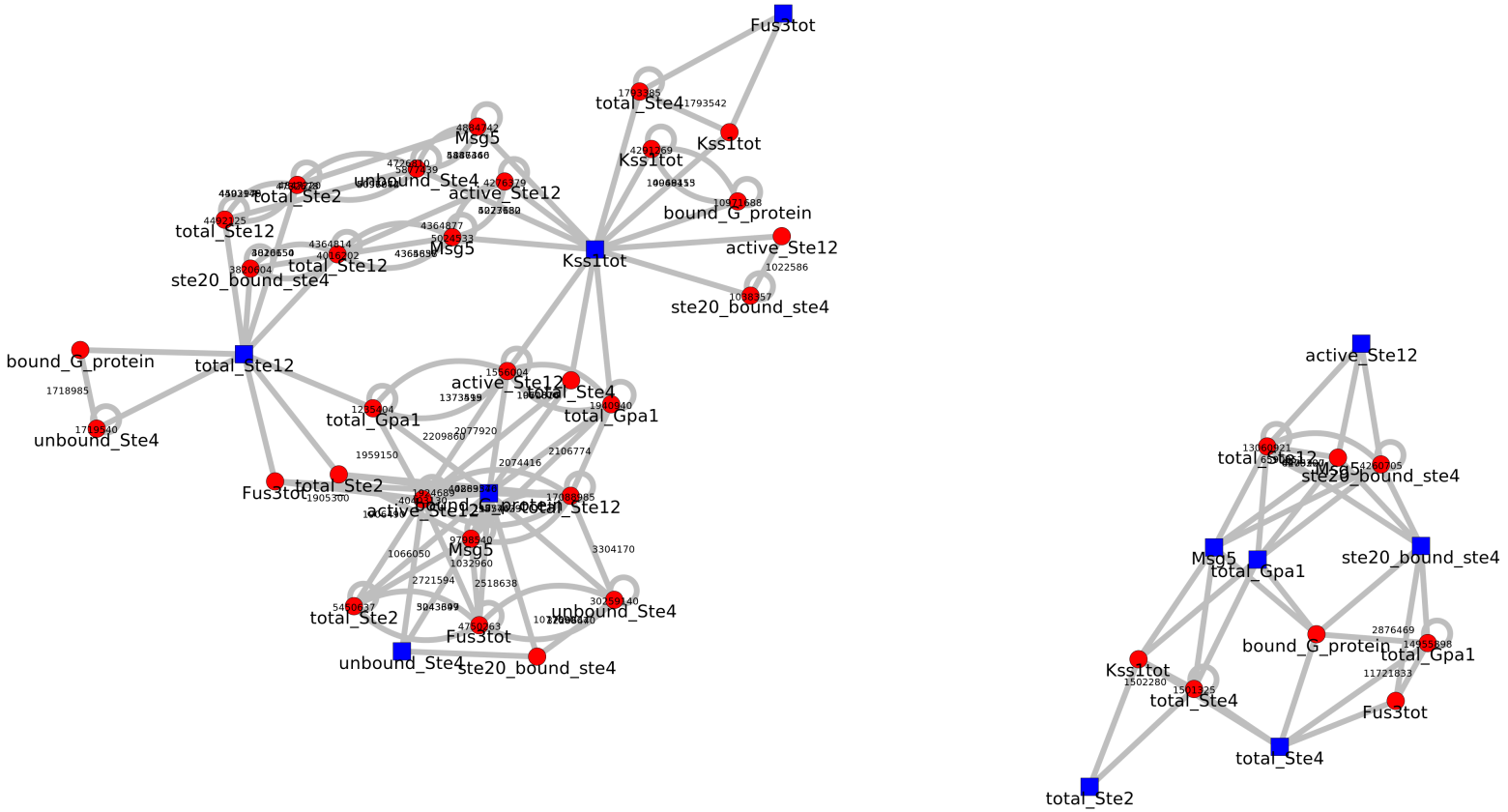


Figure 11: An example of two flux map subgraphs mapping observables to related rules. The perturbed dataset with weights > 1,000,000, blue nodes denote observables, red nodes rule names.

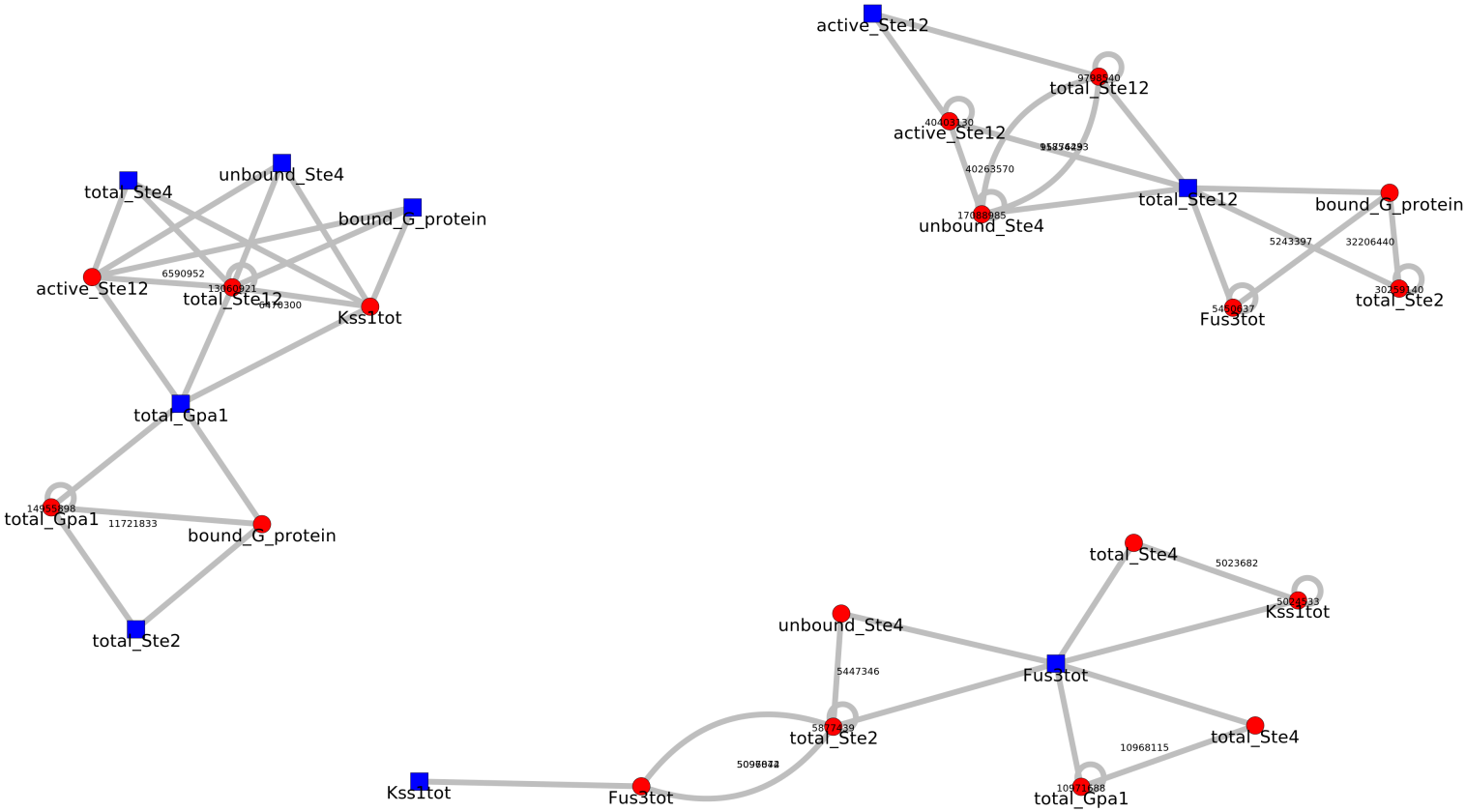


Figure 12: An example of two flux map subgraphs mapping observables to related rules. The perturbed dataset with weights > 5,000,000, blue nodes denote observables, red nodes rule names.

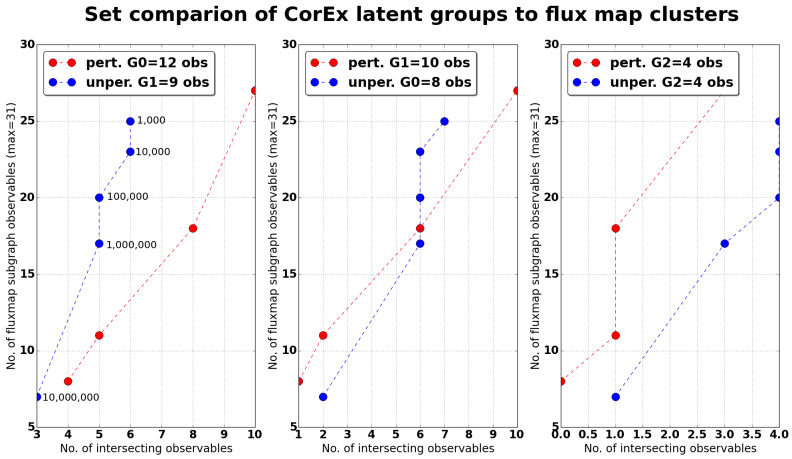


Figure 13: Three top-scored groups of latent variables (G0, G1, G2) found with CorEx, both for perturbed (red) and unperturbed (blue) simulations and five flux map subgraphs with weights above (starting from left on x-axis) 10 000 000, 1 000 000, 100 000, 10 000, 1 000 units (two last sets in the perturbed set overlap). The comparison of changes in the number of intersecting observables with decrease of stringency in rule importance shows that perturbed system gives seemingly higher overlap between compared groups than the non-perturbed dataset.

only the part after the stimulation. Furthermore, to cap the computation time, we cut the data from original 10,801 (three events per second) observations to 3,600 (one event per second).

First we examined our data by creating recurrence plots for dynamical systems [5]. The recurrence plot is an array of dots in a NxN square, where a dot is placed at (i, j) whenever $x(j)$ is sufficiently close to $x(i)$. For the purposes of this study we selected an embedding dimension of 10 and time delay 5 to keep the computational time within reasonable limits.

In general, the recurrent plot shows the times at which a phase space trajectory visits roughly the same area in the phase space [21]. The authors [5] defined small and large scale patters, textures and topologies respectively, to ease their interpretation.

The resulting figures [16 & 15] are densely grey without distinctive textures or patterns. However, the unperturbed set is seemingly brighter away from the diagonal and distinctively darker along it. This gradient is interpreted as the occurrence of a progressive decorrelation at large time intervals involving a linear trend or drift. The perturbed model presents dynamics pushed a bit more towards randomness.

Next, we created plots, showing the average mutual information index (AMI) of a given time series for a specified number of lags [8].

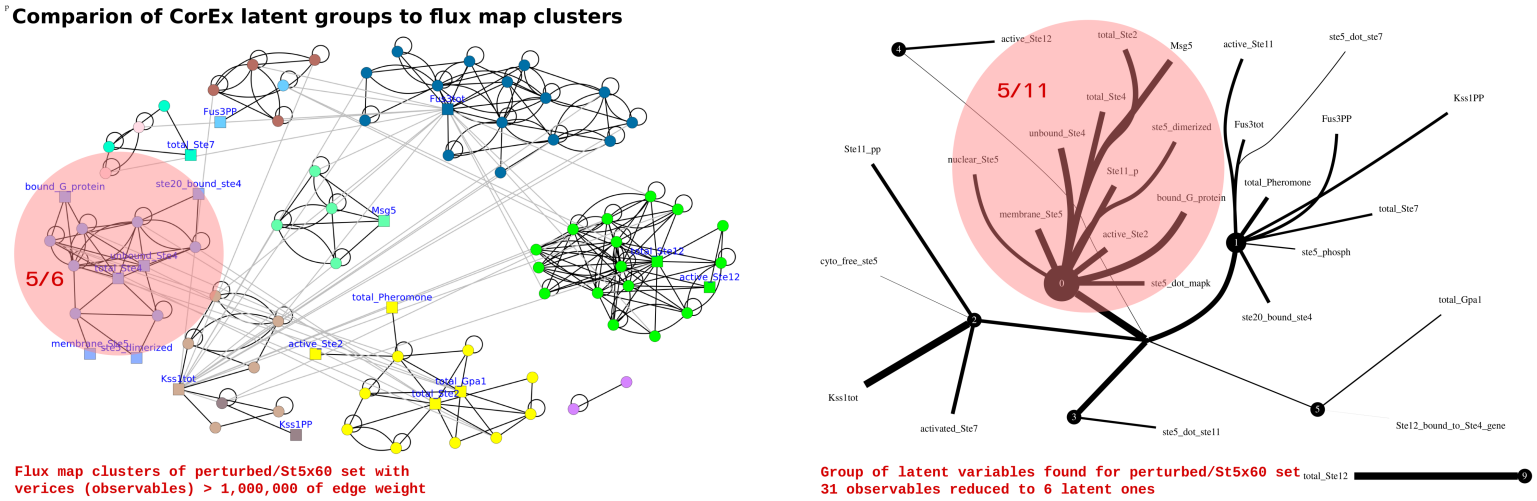


Figure 14: The largest intersection between the network of most frequently executed rules and the latent groups is apparently more visible in the perturbed model, which confirms a lack of coherence in species behaviour.

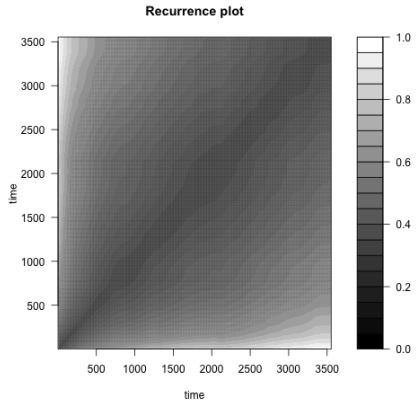


Figure 15: Unperturbed model

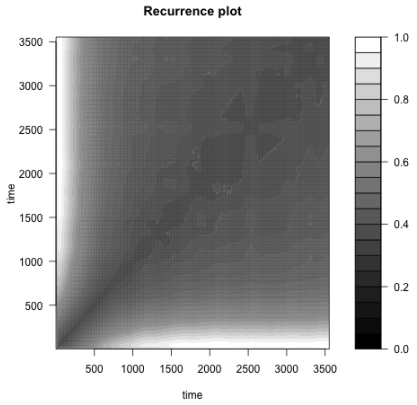


Figure 16: Perturbed model

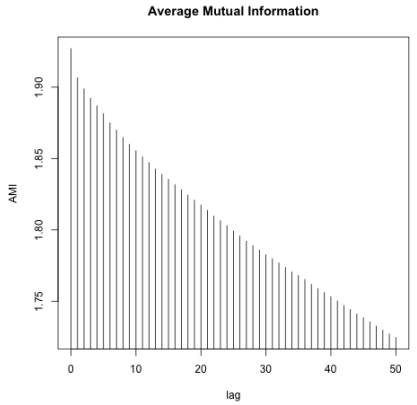


Figure 17: Unperturbed data

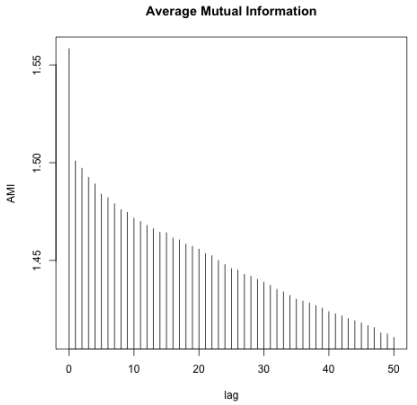


Figure 18: Perturbed model

The larger time lag the smaller is the value of AMI. In case of Figure 19 AMI drops down almost diagonally, as opposed to the Figure 20. Thus, the perturbed model is far more unpredictable, showing randomized dynamics and less interdependent relation between events.

$$S = -\sum_{ij} p_{ij}(\tau) ln \frac{p_{ij}(\tau)}{p_{ij}} \tag{6}$$

Next, we created a sample correlation integral plot [11]. The correlation integral can be approximated by the correlation sum. The correlation sum counts the number of pairs $(\overrightarrow{x}(i), \overrightarrow{x}(j))$ in a given set of vectors that are at most ϵ apart.

$$C(\epsilon) = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \Theta(\epsilon - \| \overrightarrow{x}(i) - \overrightarrow{x}(j) \|), \overrightarrow{x}(i) \in \mathbb{R}^m \tag{7}$$

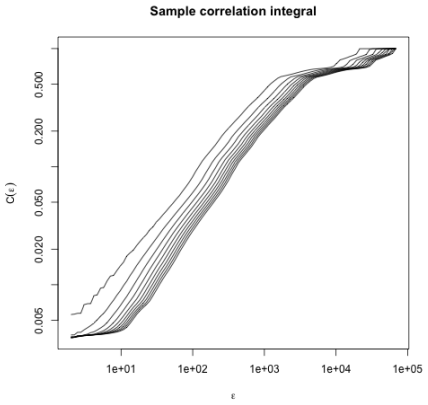


Figure 19: Unperturbed data

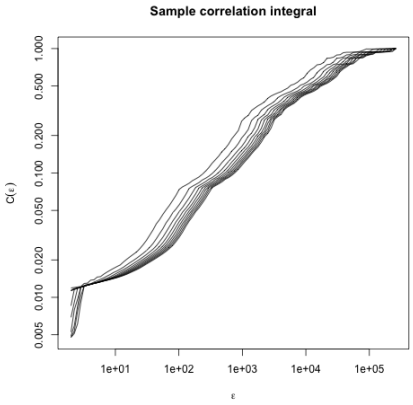


Figure 20: Perturbed data

As the perturbation involved a single parameter and demonstrated naturally occurring phenomenon (not randomised), differences between these two plots were not expected to be extreme. Nonetheless, the results are coherent both with the understanding of process and the CorEx algorithm. However, for our purposes, these methods present a more distanced view on the system dynamics, missing a decoupling problem of individual species relations.

4 Conclusions

Overall, this project offered a fruitful chance for an exploration of multivariate time series analysis. We have learned that the approach offered by the CorEx method might be very promising in analysis of rule-based models. However, it requires further testing with models that incorporate multiple randomly modified parameters and represent larger advanced processes. Further, we applied some nonlinear time series methods to our dataset. Though powerful, they offered a bird's-eye view understanding of system dynamics missing species-related details. However, both methods correctly interpreted the process offering a useful insight otherwise inaccessible.

References

[1] Lee Bardwell, Xiufen Zou, Qing Nie, and Natalia L Komarova. Mathematical models of specificity in cell signaling. *Biophysical journal*, 92(10):3425–41, May 2007.

[2] William S. Chylek, Lily A. and Harris, Leonard A. and Tung, Chang-Shung and Faeder, James R. and Lopez, Carlos F. and Hlavacek. Rule-based modeling: a computational approach for studying biomolecular site dynamics in cell signaling systems. *Wiley interdisciplinary reviews. Systems biology and medicine*, 6(1):13–36, September 2014.

[3] Vincent Danos. *Rule-Based Modelling of Cellular Signalling*, volume 4703. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[4] Gauri Dixit, Joshua B Kelley, John R Houser, Timothy C Elston, and Henrik G Dohlman. Cellular noise suppression by the regulator of G protein signaling Sst2. *Molecular cell*, 55(1):85–96, 2014.

[5] Jean-Pierre Eckmann, S Oliffson Kamphorst, and David Ruelle. Recurrence plots of dynamical systems. *Europhys. Lett*, 4(9):973–977, 1987.

[6] James R Faeder, Michael L Blinov, and William S Hlavacek. Rule-based modeling of biochemical systems with BioNetGen. *Methods in molecular biology (Clifton, N.J.)*, 500:113–67, January 2009.

[7] Jérôme Feret and Jean Krivine. *KaSim3 reference manual*, 2012.

[8] Andrew M Fraser and Harry L Swinney. Independent coordinates for strange attractors from mutual information. *Physical review A*, 33(2):1134, 1986.

[9] Shuyang Gao, Greg Ver Steeg, and Aram Galstyan. Efficient estimation of mutual information for strongly dependent variables. In *AISTATS'15*, 2015.

[10] Daniel T Gillespie. Stochastic simulation of chemical kinetics. *Annual review of physical chemistry*, 58:35–55, January 2007.

[11] Rainer Hegger, Holger Kantz, and Thomas Schreiber. Practical implementation of nonlinear time series methods: The tisean package. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 9(2):413–435, 1999.

[12] William S Hlavacek, James R Faeder, Michael L Blinov, Richard G Posner, Michael Hucka, and Walter Fontana. Rules for modeling signal-transduction systems. *Science's STKE : signal transduction knowledge environment*, 2006(344):re6, July 2006.

[13] Justin S. Hogg, Leonard A. Harris, Lori J. Stover, Niketh S. Nair, and James R. Faeder. Exact Hybrid Particle/Population Simulation of Rule-Based Models of Biochemical Systems. *PLoS Computational Biology*, 10(4):e1003544, April 2014.

[14] Bo Jin and Xinghua Lu. Identifying informative subsets of the Gene Ontology with information bottleneck methods. *Bioinformatics (Oxford, England)*, 26(19):2445–51, October 2010.

[15] Agnes Kohler, Jean Krivine, and Jakob Vidmar. A Rule-Based Model of Base Excision Repair. 2014.

[16] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review E*, 69(6):066138, 2004.

[17] A Levchenko, J Bruck, and P W Sternberg. Scaffold proteins may biphasically affect the levels of mitogen-activated protein kinase signaling and reduce its threshold properties. *Proceedings of the National Academy of Sciences of the United States of America*, 97(11):5818–5823, 2000.

[18] Bing Liu and P S Thiagarajan. Modeling and analysis of biopathways dynamics. *Journal of bioinformatics and computational biology*, 10(4), 2012.

[19] Carlos F Lopez, Jeremy L Muhlich, John A Bachman, and Peter K Sorger. Programming biological models in Python using PySB. *Molecular systems biology*, 9:646, January 2013.

[20] Abhishek Majumdar, Stephen D Scott, Jitender S Deogun, and Steven Harris. Yeast pheromone pathway modeling using Petri nets. *BMC bioinformatics*, 15 Suppl 7(Suppl 7):S13, January 2014.

[21] N. Marwan. A historical review of recurrence plots. *The European Physical Journal Special Topics*, 164(1):3–12, 2008.

[22] Dávid Pál, Barnabás Póczos, and Csaba Szepesvári. Estimation of rényi entropy and mutual information based on generalized nearest-neighbor graphs. In *Advances in Neural Information Processing Systems*, pages 1849–1857, 2010.

[23] Shirley Pepke, Greg Ver Steeg, and Aram Galstyan. Using Total Correlation Explanation to Identify Cancer Therapeutic Targets. 2015.

[24] Ryan Suderman and Eric J. Deeds. Machines vs. Ensembles: Effective MAPK Signaling through Heterogeneous Sets of Protein Complexes. *PLoS Computational Biology*, 9(10):1–35, 2013.

[25] Taiji Suzuki, Masashi Sugiyama, Jun Sese, and Takafumi Kanamori. Approximating mutual information by maximum likelihood density ratio estimation. *FSDM*, 4:5–20, 2008.

[26] Monica Duffy Toft and Yuri M Zhukov. Islamists and nationalists: Rebel motivation and counterinsurgency in russia's north caucasus. *American Political Science Review*, 109(02):222–238, 2015.

[27] Greg Ver Steeg and Aram Galstyan. Discovering structure in high-dimensional data through correlation explanation. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 577–585. Curran Associates, Inc., 2014.

[28] Greg Ver Steeg and Aram Galstyan. Maximally informative hierarchical representations of high-dimensional data. In *AISTATS'15*, 2015.

[29] Richard C Yu, C Gustavo Pesce, Alejandro Colman-Lerner, Larry Lok, David Pincus, Eduard Serra, Mark Holl, Kirsten Benjamin, Andrew Gordon, and Roger Brent. Negative feedback that improves information transmission in yeast signalling. *Nature*, 456(7223):755–761, 2008.

Appendix D

Analysis of proteins in computational models of synaptic plasticity

Authorship contribution

Work on this project started before I joined it. The main concept and design of this study belong to the other authors. Together with K.F.Heil and D.C.Sterratt, we selected models, identified their constituent entities and mapped them to reference identifiers. To my contribution belongs a review of the models of striatal synaptic signalling pathways. I summarised and described sources of selected models. I performed enrichment analysis of modelled genes and provided results analysis. I contributed to proofreading and reviewing the manuscript.

Analysis of proteins in computational models of synaptic plasticity

Katharina F. Heil[1,2☯], Emilia M. Wysocka[1☯], Oksana Sorokina[1], Jeanette Hellgren Kotaleski[2], T. Ian Simpson[1], J. Douglas Armstrong[1], David C. Sterratt[1*]

1 School of Informatics, University of Edinburgh, Edinburgh, Scotland, UK
2 Computational Science and Technology, School of Computer Science and Communication, KTH Royal Institute of Technology, Stockholm, Sweden

☯These authors contributed equally to this work.
* david.c.sterratt@ed.ac.uk

Abstract

The desire to explain how synaptic plasticity arises from interactions between ions, proteins and other signalling molecules has propelled the development of biophysical models of molecular pathways in hippocampal, striatal and cerebellar synapses. The experimental data underpinning such models is typically obtained from low-throughput, hypothesis-driven experiments. We used high-throughput proteomic data and bioinformatics datasets to assess the coverage of biophysical models.

To determine which molecules have been modelled, we surveyed biophysical models of synaptic plasticity, identifying which proteins are involved in each model. We were able to map 4.2% of previously reported synaptic proteins to entities in biophysical models. Linking the modelled protein list to Gene Ontology terms shows that modelled proteins are focused on functions such as calmodulin binding, cellular responses to glucagon stimulus, G-alpha signalling and DARPP-32 events.

We cross-linked the set of modelled proteins with sets of genes associated with common neurological diseases. We find some examples of disease-associated molecules that are well represented in models, such as voltage-dependent calcium channel family (*CACNA1C*), dopamine D1 receptor, and glutamate ionotropic NMDA type 2A and 2B receptors. Many other disease-associated genes have not been included in models of synaptic plasticity, for example catechol-O-methyltransferase (*COMT*) and *MAOA*. By incorporating pathway enrichment results, we identify *LAMTOR*, a gene uniquely associated with Schizophrenia, which is closely linked to the MAPK pathway found in some models.

Our analysis provides a map of how molecular pathways underpinning neurological diseases relate to synaptic biophysical models that can in turn be used to explore how these molecular events might bridge scales into cellular processes and beyond. The map illustrates disease areas where biophysical models have good coverage as well as domain gaps that require significant further research.

Author summary

The 100 billion neurons in the human brain are connected by a billion trillion structures called synapses. Each synapse contains hundreds of different proteins. Some proteins sense the activity of the neurons connecting the synapse. Depending on what they sense,

the proteins in the synapse are rearranged and new proteins are synthesised. This changes how strongly the synapse influences its target neuron, and underlies learning and memory. Scientists build computational models to reason about the complex interactions between proteins. Here we list the proteins that have been included in computational models to date. For good reasons, models do not always specify proteins precisely, so to make the list we had to translate the names used for proteins in models to gene names, which are used to identify proteins. Our translation could be used to label computational models in the future. We found that the list of modelled proteins contains only 4.2% of proteins associated with synapses, suggesting more proteins should be added to models. We used lists of genes associated with neurological diseases to suggest proteins to include in future models.

Introduction

Activity-dependent synaptic plasticity is necessary for learning and memory [1]. Since the discovery of long term potentiation (LTP) and long term depression (LTD) [2, 3], it has been shown that synaptic plasticity can depend strongly on patterns of pre-and post-synaptic firing [4] and neuromodulators [5]. Forms of plasticity vary between types of synapses and brain region [4], which could be explained by the local proteome, i.e. the expressed proteins and their abundances; PSD-95 knock-outs demonstrate the influence of the proteome on synaptic plasticity [6]. Synaptic plasticity underlies behaviour, as evidenced by the effect of antagonising NMDA receptors [1], and synaptic proteins underlie disease [7].

Synapses have been modelled computationally at various levels of detail. Models at a phenomenological level, such as spike-timing dependent plasticity (STDP) models, link firing patterns in the pre- and postsynaptic neurons to changes in synaptic strength with little or no reference to the underlying molecules [8]. Biophysical models refer to at least some known molecular actors in synaptic plasticity. In 2009 there were at least 117 biophysical postsynaptic signal transduction models [9] and the number is growing [10, 11].

Recent advances in tissue and cell extraction techniques and sample processing allow localised proteomes to be determined, e.g. the synapse including the smaller presynaptic or postsynaptic proteomes [12, 13]. The most recent analysis of 37 published synaptic proteomic datasets contains 1,867 presynaptic genes, 5,053 postsynaptic genes and 5,862 synaptic genes (with human EntrezID identifiers) respectively. These numbers are large compared to results from individual studies. Nevertheless, data inclusion was highly restrictive and the augmented numbers can be partly explained by higher experimental sensitivity and the broad use of high-throughput techniques (a manuscript containing detailed analysis of the synaptic proteome is in preparation).

These synaptic protein lists make it possible to compare systematically proteins contained in computational models of synapses with those proteins likely to be in the synapse. In this paper we: (1) survey a selection of biophysical models of synaptic plasticity, identifying which proteins are involved in each model, and describing the complexity and detail of description of signalling pathways within the models; (2) compare the proteins in models with synaptic protein lists, thus showing what fraction of synaptic proteins have been considered in models; (3) identify the functional classes of proteins in models; and (4) compare the proteins in models with those involved in neurological diseases. This work should help inform what proteins and pathways should be considered in new modelling efforts. While new datasets offer possibilities for models of greater scope and detail, it is important to understand the foundations that have been laid by existing computational models of synaptic plasticity, which we do thematically before moving to the identification of proteins in models and the discussion

of implications of our findings for future synaptic models and model annotation.

Biophysical models of synaptic plasticity

To set the scene for our analysis of proteins in biophysical models of synapses, we first give an overview of how the questions addressed in models of synaptic plasticity have shaped the development of simulation methods, and describe the main hippocampal, striatal and cerebellar pathways that have been modelled. We categorise simulation methods as non-spatial, spatial or multiscale and as deterministic or stochastic. Table 1 shows examples of simulation packages and associated studies that fall into each category. Rather than using simulators, some studies use bespoke code in languages such as Java, or generic mathematical environments such as MatLab.

Table 1. Overview of simulation environments.

	Deterministic	**Stochastic**
Non-spatial	Berkeley Madonna 8.0 (BM8) [14][1] GENESIS [15][2] Java [16][3]) ode15s (SimBiology, MatLab toolbox) [17, 18][4] PLAS (Power Law Analysis and Simulation) [19][5] Xcellarator (Mathematica) [20][6] XPPAUT [21, 22][7]	KaSim [23][8] StochSim [24][9]
Spatial	NEURON[10] STEPS [25][11] Virtual Cell [26][12]	MCell [27][13] NeuroRD [28–31][14] Smoldyn [32][15] STEPS [25]
Multiscale	E-Cell 3 ode [33][16] NEURON + E-Cell 3 ode [34]	

Simulation environments listed according to whether they support deterministic or stochastic simulations, and non-spatial, spatial or multiscale simulations (ordered alphabetically by simulator). URLs for the simulation environments are indicated by superscripts (see below). References for studies using each simulation are given.
[1]`http://www.berkeleymadonna.com/index.html` [2]`http://www.genesis-sim.org/`
[3]`https://www.java.com/en/` [4]`http://uk.mathworks.com/products/simbiology/`
[5]`http://enzymology.fc.ul.pt/software/plas/` [6]`http://www.cellerator.info/`
[7]`http://www.math.pitt.edu/~bard/xpp/xpp.html`
[8]`https://github.com/Kappa-Dev/KaSim/`
[9]`https://sourceforge.net/projects/stochsim/`
[10]`https://www.neuron.yale.edu/neuron/`
[11]`http://steps.sourceforge.net/STEPS/default.php` [12]`http://vcell.org/`
[13]`http://mcell.org/` [14]`http://krasnow1.gmu.edu/CENlab/software.html`
[15]`http://www.smoldyn.org/about2.html` [16]`http://www.e-cell.org/`.

Non-spatial models

Many of the simulation methods and issues associated with models of signalling pathways are found in models of calcium/calmodulin dependent kinase II (CaMKII)

and the intricate dynamics of its phosphorylation states and interactions with calcium-bound calmodulin (CaM).

Mean field models of CaMKII In 1985 Lisman [35] advanced the hypothesis, expressed as a mathematical model, that memories could be stored in bistable molecular switches comprised of auto-phosphorylating kinases. Following the discoveries that CaMKII is an autophosphorylating holoenzyme [36] and is a major component of the postsynaptic density (PSD) [37], Lisman and Goldring [38] proposed that CaMKII could form the basis for the auto-phosphorylating switch. Their ordinary differential equations (ODEs) described how the probability of a CaMKII holoenzyme being "on" – the "mean field" – could depend on the calcium concentration and the number of phosphorylation sites required to switch the CaMKII holoenzyme on. Solving these equations demonstrated that the number of CaMKII holoenzymes activated could depend on the duration of the calcium stimulus, thus allowing CaMKII to act as graded rather than binary switch. Furthermore, the time taken for the switch to turn on could be modulated by changing the threshold number of sites that needed to be phosphorylated before the holoenzyme entered an auto-phosphorylated state.

Analysis of mean field models Mean-field ODE models allow stability analysis to be undertaken, which can show, for example, that a model of CaMKII has two stable states – almost fully phosphorylated or almost fully dephosphorylated – within a wide range of calcium concentrations [39]. Stability analysis has also been used to inform how parameters should be set to give a biphasic calcium-synaptic strength curve, with LTD at moderate concentrations of calcium and LTP at high concentrations [22].

Stochastic models of CaMKII In a volume containing N reacting molecules of a species, there will be fluctuations of the order of $1/\sqrt{N}$ in the concentration of the species predicted by the mean-field solution. For large volumes it follows that stochastic effects can be neglected, but in the $\sim$1 fl volume of the spine head the number of CaMKII holoenzymes is considerably finite – an average of 30 are seen in electron microscopy (EM) images of immunuogold labelled PSDs [40] – so there will be significant variability between experiments in the same conditions. In order to determine the accuracy of the encoded information for a given number of holoenzymes, Lisman and Goldring [38] used the binomial formula to compute the mean and standard deviation of the number of fully phosphorylated CaMKII holoenzymes, which suggested that graded information could be stored to an accuracy of around 10%.

Rather than deriving variability from mean field simulations, stochastic ("Monte-Carlo") models can be built. Each run of a stochastic model is generated by drawing random numbers to decide when bonds are made or broken, and when changes in state occur; the variability of the model is obtained by analysing multiple runs. A simple method to simulate chemical reactions accurately is Gillespie's stochastic simulation algorithm (SSA) [41], as used in some simulations [42].

Combinatorial complexity in models of CaMKII One challenge in modelling CaMKII is that each CaMKII holoenzyme comprises multiple subunits; initial estimates were of 8–14 subunits, but EM and X-ray crystallography show that there are 12 subunits [43–45] arranged in two hexamer rings. Since a phosphorylated subunit can act as a kinase to its neighbour, the multiple subunits give rise to a combinatorially large number of meaningful configurations (states) of the holoenzyme. For example, a model with 6 subunits, each of which can be in one of 12 states, can be in 498,004 configurations according to the necklace function [46] and would therefore need the

same number of ODEs to simulate. To simulate the dodecamer ring would require $\sim 10^{12}$ states, an impractical number of states to model with ODEs.

This combinatorial problem can be alleviated by model simplification, for example by (i) reducing the number of subunits to 4 and (ii) lumping together states that are invariant to rotations and adjusting the reaction rates between states according to their multiplicities [47]. These strategies are used in other deterministic and stochastic models of CaMKII [22, 48, 49]. A further simplification can be made by lumping together states with the same number of phosphorylated subunits, and weighting the transition rates between these states [39].

Agent-based simulation Combinatorial complexity can also be dealt with using agent-based simulation, in which the states of individual molecules rather than populations of molecules are followed through the simulation [50]. For example, in simulations of a 10-subunit CaMKII holoenzyme [51], there was one variable per subunit, each of which described which of 5 states the subunit was in. The state of each holoenzyme was therefore described by 10 state variables, giving 976,887 states of the holoenzyme. Transition probabilities between a subunit's states depended on its own state and that of its neighbouring subunit. Transitions were generated in 100 ms time steps in each subunit in turn, based on the state of the holoenzyme in the previous time step – similar to the τ-leap algorithm later formalised by Gillespie [52]. As this method is based on a fixed time step it can be combined with deterministic simulation of some elements of the system, as in a model of CaMKII activation in a dendritic spine [53].

Rule-based simulation Agent-based simulation alone does not solve the problem of how to represent the states and the transitions between states clearly and concisely [50]. To specify transitions in agent-based simulations "rules" are specified in which the state of a fragment of system is mapped to the transitions that can occur within that fragment. For example a CaMKII monomer may be phosphorylated when both it and its neighbour (the fragment) are bound to Ca^{2+}-CaM complex [24]. The StochSim agent-based simulator [54] describes rules by using flags to represent phosphorylation and binding states to be attached to molecules. However, the StochSim description of binding of CaM to CaMKII, phosphorylation states of CaM and trapping of CaM by CaMKII [24] is, arguably, unwieldy, requiring 1,209 lines of code.

Second generation rule-based modelling languages such as Kappa [55] or BioNetGen (BNGL, [56]) have a well-defined, general syntax to specify binding sites and states of proteins and interactions between protein binding domains. The interaction rules can be expanded to generate the "biological network", i.e. the full set of complexes and reactions needed to simulate the system [56]. These reactions can be converted into ODEs or stochastic differential equations (SDEs), or simulated using a stochastic simulation method [41, 52]. In another approach – dubbed "Network Free" [56], since no biological network is generated – simulators, such as KaSim [55] or NFSim [57], create the complexes that exist throughout a simulation dynamically. Network-free methods avoid the prohibitive memory requirements needed to store all possible states in a large network [57], and even allow simulations with infinite numbers of potential species [55]. This form of "on-the-fly" simulation is intrinsically stochastic, with transitions occurring one rule at time, similar to Gillespie's SSA [41]. For smaller networks, ODEs, SDEs or the SSA are faster, but because the simulation speed of these methods scales roughly with network size (i.e. the number of reactions), for larger networks these conventional methods are slower than network-free simulation [57].

Varying model structures Authors devise differing descriptions of the same pathway. For example Byrne et al. [58], Stefan et al. [59] and Faas et al. [14] all describe

the binding of calcium to CaM, but each model has a distinct structure. The models of Byrne et al. and Faas et al. assume cooperativity within the N and C lobes of CaM: the rate at which a calcium ion binds to a lobe with one calcium bound is different from the rate at which calcium binds to the lobe in the *apo*, unliganded, state. In contrast, Stefan et al. assume that the affinity of each of the four positions on CaM is independent, but that these affinities depend on whether the entire CaM molecule in the "tense" or "relaxed" conformation [60], which is an allosteric mechanism [61]. The two positions within each lobe are assumed to be equivalent by Faas et al., but not by Byrne et al. The model of Faas et al. has been fit against kinetic data, which is richer than the binding curves fit by Byrne et al. and Stefan et al., but it has not been investigated whether the parameters of these earlier models could be adjusted to fit the kinetic data.

There is also diversity in the number of states monomers in models of CaMKII may assume, and how the multimeric structure of the molecule is represented. An additional variation in particle-based simulations of CaMKII is that once the CaM N or C lobe is bound to a CaMKII monomer, it becomes much more likely that the other lobe on the same CaM molecule will bind to a neighbouring CaMKII monomer on the hexamer ring [58]. This necessary to fit Ca-chelator-induced dissociation curves [62] and steady-state CaM-CaMKII binding curves [43]. A result of this assumption is that the rate of CaM binding to CaMKII is dominated by the more affine N-lobe.

Biophysical constraints on parameters A number of strategies are used to reduce the considerable number of reaction coefficients in molecular models. For example, the reactions in Byrne et al. [58] are parameterised by 2 sets of 24 parameters, but the forward reaction coefficients are all set to be equal, reducing the number to 2 sets of 13. The principle of microscopic reversibility [61] is used to link reaction coefficients that are in loops, taking the number down to 2 sets of 9. Microscopic reversibility applies generally, though some ion channels are exceptions to this rule [61]. Other linkages between parameters can be postulated; for example in the allosteric model of Stefan et al. [59], the ratio between the affinities of each site for calcium in the tense and relaxed conformations is assumed to be the same for each of the four sites.

Data used to constrain parameters Various types of data have been used to constrain the parameters of single pathway models. To obtain equilibrium binding curves, equilibrium dialysis with radioactively labelled ligands can be used, as by Crouch and Klee in their determination of Ca^{2+}-CaM binding. More recently, stopped-flow fluometry [43] has been used for the same purpose. This method has the disadvantage of a relatively long dead time of the order of 2 ms, which hinders determining fast dynamics, e.g. of the N lobe of CaM. A faster method is calcium uncaging, which can lead to a sub-0.1 ms change in calcium concentration, and measurement with a fast fluorescent calcium indicator [14].

Spectroscopic analysis can be used to infer conformational changes, e.g. the tense to relaxed conformation change upon binding of a calcium ion to CaM [60]. Phosphorylation states, e.g. of CaMKII, can be measured using radioactively labelled ATP [43] which can be coupled with immunoprecipitation and gel electrophoresis [63].

Optimisation of free parameters Even after reducing the number of parameters there are typically a number of free parameters in a model, and a number of optimisation techniques are used to fit them to data, for example particle swarm optimisation [58]. Latin hypercube sampling can be used to determine global parameter sensitivity [20].

Hypothesis-driven and simplified modelling In one combined experimental-modelling study [63], the authors engineered a monomeric form of CaMKII. This allowed them to measure the CaM-dependent phosphorylation properties of CaMKII and produce a simplified computational model, which predicted that the amount of CaMKII activation would depend on the frequency of a presented train of Ca pulses: CaMKII could thus act as a frequency decoder. A number of CaMKII models at various levels of detail have been formulated to explain the dependence of CaMKII activation on the frequency of calcium pulses [47, 48, 64].

Data-driven rule-based modelling Proteomic studies of the synapse (Table S2) show that there are many proteins in the synapse not included in the models described thus far. The challenges of combinatorial complexity, already encountered in models of CaMKII, are magnified as more proteins are added. Rule-based modelling has been applied to simulate a network containing 54 proteins, with interactions were described by 136 rules [23]. This model makes predictions about the molecular composition of complexes that could occur in the PSD.

Spatial models

The modelling methods described so far assume that molecules are within a well-stirred, spatially homogeneous environment. However, the cellular environment is not homogeneous; for example, calcium enters through N-methyl-O-aspartic acid receptors (NMDARs) on one side of the spine head. It can react with buffers on a shorter timescale than it takes to diffuse through the spine, and can exist within microdomains around the NMDARs briefly at high concentrations. Thus, to address some questions, it is necessary to model space explicitly.

Deterministic reaction-diffusion Deterministic diffusion is modelled by splitting cellular space into compartments and formulating ODEs to describe how reactions within compartments and fluxes between compartments affect the concentrations of species within each compartment. Deterministic diffusion along one dimension has been used in models of calcium and other intracellular signalling in spines [65–67]. Whilst these models do not model LTP and LTD explicitly, they give insights such as that the combination of calcium pumps and buffers can confine calcium and activated CaMKII to the synaptic spine head [67], or that the temporal ordering of input at weak and strong synapses with NMDARs determines the concentration of calcium in the spine, which will then influence the intracellular pathways underlying LTP and LTD [66]. The NEURON simulator, used widely in models of electrical activity of neurons, also supports reaction-diffusion, with recent work to extend these capabilities [68]. Deterministic reaction-diffusion can be simulated in 3D by splitting cellular space into tetrahedral or cubic compartments, as implemented in the STEPS simulator [69].

Compartmental stochastic reaction-diffusion The numbers of molecules in each compartment of a mesh is often small enough to warrant stochastic simulation methods. Gillespie's SSA can be extended to a compartmentalised volume by replicating the set reactants in each compartment, and treating diffusion of reactants between each compartment as a type of reaction [41]. This "Spatial SSA" method and more efficient approximations [70] have been used for a number of simulations of medium spiny projection neurons in the striatum [28–31] and is implemented in the simulators NeuroRD [28] and STEPS [69].

Compartmental agent-based stochastic reaction-diffusion The Spatial SSA requires one variable in each compartment to describe the number of molecules in every possible state in the system, and therefore is ill-adapted to deal with models of molecules with many states, such as CaMKII. A custom extension to the Spatial SSA has been used to study the relative effects of the stochastic opening and closing of NMDARs and of stochastic binding between CaMKII holoenzymes and CaM in a spine head [27]. The results showed that NMDARs were a greater source of noise, due to their smaller numbers than the CaMKII holoenzymes. The agent-based, rule-based simulator SpatialKappa [71] extends the Kappa language syntax and the KaSim algorithm to allow diffusion of complexes between voxels in regular meshes.

Particle-based stochastic reaction-diffusion In particle-based simulation methods, each molecule has a location in 3D space or on a 2D membrane and moves in Brownian leaps. Reactions may occur when particles come within an interaction radius of each other. Simulators implementing this method include MCell [72] and Smoldyn [73]. MCell has been used to model diffusion of glutamate molecules in the synaptic cleft and their binding to NMDARs and α-amino-3-hydroxy-5-methyl-4-isoxalone propionic acid receptors (AMPARs) [27, 74], and influx of calcium into the spine head and its interaction with calcium binding proteins [75, 76]. The most recent version of Smoldyn supports the rule-based BNGL language, but only to generate reaction networks, not to perform network-free simulation.

Modelling diffusion measurements Khan et al. [32] used a spatial model built with Smoldyn to interpret their fluorescence recovery after photo-bleach (FRAP) measurements of CaMKII diffusing in a spine head before and after glutamatergic stimulation. Eleven bidirectional reactions described binding of phosphorylated CaMKII to the PSD, binding of non-phosphorylated CaMKII to the actin cytoskeleton, and CaMKII self-aggregation. All these reactions contribute to keeping stable CaMKII concentrations in stimulated spines, providing an explanation of sequestration of CaMKII in dendritic spines.

Multiscale modelling It is possible to simulate reaction-diffusion and the membrane potential using the same spatial mesh, but these simulations are likely to run very slowly because of the unnecessarily fine mesh in parts of the model, such as the dendrites, where concentration gradients are lower. Multiscale modelling, defined as the process of using multiple models at different scales simultaneously to describe a system [77], can allow for the desired level of detail with tractable simulation times. To demonstrate a multiscale algorithm to integrate detailed models of signalling networks within electrical models of neuron, Mattioni and Le Novère [34] used a model of a striatal medium spiny projection neuron (MSPN) with 1,000 synaptic spines attached. The electrical activity and calcium accumulation in the dendrites and soma of the neuron were simulated using the NEURON implementation of the compartmental modelling method. Within each spine, the calcium flux through AMPARs, NMDARs and voltage gated calcium channels (VGCCs) calculated by the electrical model is fed to instances of a molecular simulator (in this case E-CELL3), in which the calcium binds to CaM, which then participates in a biochemical network typical of striatal MSPNs. A similar effort has incorporated the rule-based SpatialKappa simulator into NEURON [78].

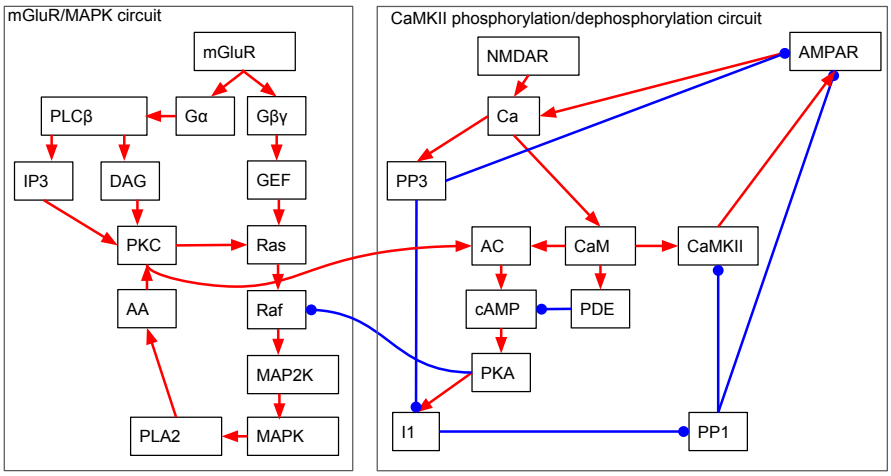


Fig 1. Partial block diagrams comparing essential elements of the hippocampal biochemical circuit. Each small box represents an ion, monomer or multimer. Red arrows indicate activating interactions. Blue lines ending in circles represent inhibiting interactions. Within each box, the molecules can be one of potentially many binding or phosphorylation states. The circuit is split into two sub-circuits: the CaMKII phosphorylation/dephosphorylation circuit and the mGluR/MAPK circuit.

Models of hippocampal synaptic signalling pathways

In tandem with the extensive experimental study of LTP and LTD in the hippocampus, computational models of hippocampal synaptic plasticity have been developed.

The CaMKII phosphorylation-dephosphorylation circuit Lisman [79] proposed a model to account for how LTP and LTD could be mediated by postsynaptic calcium acting as a second messenger (Fig 1). A high concentration of calcium, caused by coincident pre- and postsynaptic activity, leads, via binding to CaM, to phosphorylation and then auto-phosphorylation of CaMKII. At moderate concentrations calcium binds to calcineurin (PP3), which is also known as PP2B; we use PP3 for consistency with gene identifiers. The calcineurin-calcium complex dephosphorylates protein phosphatase inhibitor 1 (I1), thereby deactivating it. The inactive I1 then unbinds from protein phosphatase 1 (PP1), allowing it to dephosphorylate phosphorylated CaMKII. At high Ca^{2+} levels this pathway is inhibited via Ca^{2+}-CaM activated adenylate cyclase (AC), which then catalyses production of cyclic adenosine monophosphate (cAMP) from adenosine triphosphate (ATP). The cAMP then binds to the regulatory subunits of cAMP-dependent protein kinase (PKA), releasing its catalytic subunits which then phosphorylate I1, thereby allowing it to sequester PP1. The Ca^{2+}-CaM complex also activates phosphodiesterase (PDE), which hydrolises cAMP into adenosine monophosphate (AMP), thus reducing the rate of activation of PKA.

Lisman formulated this biochemical circuit as a simplified steady-state mathematical model of the net phosphorylation rate of CaMKII, and showed that a set of parameters existed that would allow unphosphorylated ("off") CaMKII molecules to be phosphorylated (activated) by high Ca^{2+} levels, and phosphorylated ("on") CaMKII molecules to be dephosphorylated (inactivated) by low Ca^{2+} levels. Lisman

hypothesised that, ultimately, CaMKII activation increases the non-NMDA component of the synaptic response. The biochemical circuit of Lisman is included in a number of dynamical biochemical models of postsynaptic signal transduction [80–84]. In some cases PKA is assumed to be tonically active rather than released from inhibition by cAMP, [83, 85] and other features may be included such as sequestering of CaM by neurogranin and SAP97 [83].

AMPA receptor phosphorylation Models have been formulated in response to the developing understanding of AMPARs [86]. AMPARs comprise four subunits, each of which is one of GluR1–4. The phosphorylation at two sites on GluR1 affects the function of the AMPAR multimer. In synapses in a "naive" state, i.e. those which have not been exposed to any plasticity protocols, phosphorylation of Serine 831 (Ser831), by CaMKII or protein kinase C (PKC), is associated with LTP [87, 88] and dephosphorylation of Serine 845 (Ser845) is associated with LTD [89]. In synapses that have already experienced LTD, "dedepression" caused by a theta-burst stimulus is associated with Ser845 phosphorylation, and in a synapse that has potentiated, the Ser831 site is dephosphorylated during "depotentiation" [88].

These findings led to the four state model of AMPARs by Castellani et al. [90], in which potentiation is caused by phosphorylation of the Ser831 and Ser845 sites, and LTD caused by dephosphorylation of the sites. The activation of the phosphatases and kinases was set up in the model so that the phosphates were more activated than the kinases at low concentrations, and vice-versa for high concentrations. Steady-state analysis of the set of 4 bidirectional reactions gave a typical biphasic Ca^{2+}-synaptic strength curve in which there is LTD at moderate concentrations of calcium and LTP at high concentrations. Furthermore, control of Ca^{2+} levels via adaptation of NMDARs allowed modification of the threshold level of Ca^{2+} at which LTP rather than LTD occurred, as in the Bienenstock-Cooper-Munro (BCM) rule [91].

AMPAR trafficking Blocking AMPAR exocytosis causes run-down of synaptic strengths, and inhibiting endocytosis of AMPARs causes an increase in AMPAR responses [92]. This discovery lead to the idea of a stable distribution of receptors at the synapse being replaced by a highly dynamic picture, with continuous exocytosis and endocytosis of AMPARs [93]. The trafficking to synapses comprises three steps [94]: (1) AMPARs bound to Transmembrane AMPA receptor regulatory protein (TARP) proteins such as stargazin are inserted into the dendritic shaft or spine by phosphorylation events caused by PKA, PKC, extracelluar regulated kinase (ERK) (part of the mitogen-activated protein kinase (MAPK) family) or Phosphoinositide 3-kinase (PI3K), or myosin-V; (2) the AMPARs diffuse through the membrane to the synapse; and (3) phosphorylation events (triggered by active CaMKII targeting stargazin) increase the affinity of the AMPAR-stargazin complex for PDZ-containing scaffolding proteins such as PSD95, PSD93, SAP97 and SAP102. AMPAR trafficking away from synapses is thought to be an inverse process, whereby AMPARs are released from PDZ proteins and diffuse from the synapse back to the dendrite, where they are endocytosed. There is a link between trafficking and the phosphorylation states of AMPARs, with phosphorylation of Ser845 on the GluR1 subunit needed to incorporate GluR1 subunits into synapses [95], although it is not clear how strong this link is [96].

Integrated modelling of CaMKII phosphorylation circuit and AMPAR trafficking Urakubo et al. [84] explored whether a model that integrated AMPAR trafficking with the CaMKII-phosphorylation-dephosphorylation biochemical circuit first formulated by Lisman and implemented by Bhalla and Iyengar [80] could account for spike-timing dependent plasticity (STDP). They embedded the circuit in a spine

containing NMDARs, AMPARs and VGCCs in a simplified soma-and-dendrite compartmental model with conductances used in models of CA1 hippocampal cells [97, 98]. In their first model LTP resulted from pre-before-post spiking, but LTD did not result from post-before-pre spiking. To cause LTD in this situation, it was sufficient that the NMDARs were blocked by binding of Ca^{2+}-bound CaM. This biochemical detection circuit was linked to AMPAR phosphorylation and dephosphorylation by the activities of the kinases CaMKII and PKA and the phosphatases PP1, PP3 and protein phosphatase 2 (PP2) (commonly known as PP2A). AMPAR trafficking was modelled by having four pools of AMPARs: (1) cytosolic; (2) in the dendritic or spine shaft membrane; (3) at the synapse but not anchored by PDZ proteins; and (4) at the synapse, anchored by PDZ proteins. The phosphorylated LTP and LTD states were used to control the rates of endo- and exocytosis, and binding to the PDZ proteins.

The MAPK circuit and metabotropic glutamate receptor (mGluR) signalling To the CaMKII phosphorylation-dephosphorylation circuit the modular model of Bhalla and Iyengar [80] adds the MAPK cascade, activated by mGluRs (Fig 1). Input to mGluRs activates G-proteins, which then go on to activate phospholipase C-β (PLC-β), leading to production of diacylglycerol (DAG) and inositol (IP3) and phosphorylation of PKC. This activates the cascade of Ras, Raf, mitogen-activated protein kinase kinase (MAP2K) and MAPK. In turn, MAPK activates phospholipase A2 (PLA2), which cleaves arachidonic acid (AA) from phospholipids. The AA binds to PKC, activating it, which in turn leads to more Ras activity, completing the loop. The G-proteins also activate the Ras–Raf–MAP2K–MAPK pathway via up-regulation of guanine exchange factor (GEF). The parameters in the system were such that the persistent up-regulation of PKC was enough to catalyse AC production in the CaMKII circuit, and thus up-regulate PKA and down-regulate PP1, leading to prolonged CaMKII activation. There was also inhibitory crosstalk from the CaMKII to the MAPK via inhibition of Raf by PKA.

Late LTP, synaptic tagging and gene expression The models described so far all deal with the induction of early-LTP, which occurs up to 4 hours after induction and does not depend on protein synthesis [99]. In contrast, late-LTP depends on protein and mRNA synthesis. In order to solve the conundrum of how AMPAR proteins, which were assumed not to be synthesised close to synapses, get to the synapses, Frey and Morris [99] proposed that a "synaptic tag" is set when activity has potentiated the synapse. Smolen et al. [100] formalised this concept into an ODE model containing four pathways: (1) the MAPK cascade; (2) PKA activated by cAMP; (3) CaMKII; and (4) Ca^{2+}-activated calcium/calmodulin-dependent protein kinase kinase (CaMKK), which activates calcium/calmodulin-dependent protein kinase (CaMKIV). The CaMKII, MAPK, and PKA pathways are all required to set a synaptic tag. CaMKIV, assumed to be in the nucleus, and MAPK are assumed to activate unknown transcription factors. The input to the model was the assumed time courses of Ca^{2+}, Raf and cAMP. The CaMKII phosphorylation circuit was not modelled.

To induce late-LTP, translation and synaptic tags need to be active simultaneously. Smolen et al. [16] devised a distinct model at a similar, relatively low, level of detail containing notional synaptic LTP tags activated by Ca^{2+}-CaM-CaMKII, LTD tags activated by the Raf-MAPK pathway, local protein translation mediated by autonomously active isoform of atypical protein kinase C ζ (PKMζ) (after a chequered history, back in favour as a memory molecule [101]), and movement of PKMζ and notional plasticity related proteins from the cytoplasm to synapses. The model was used to explore how strong potentiating or depressing stimuli at one synapse can promote

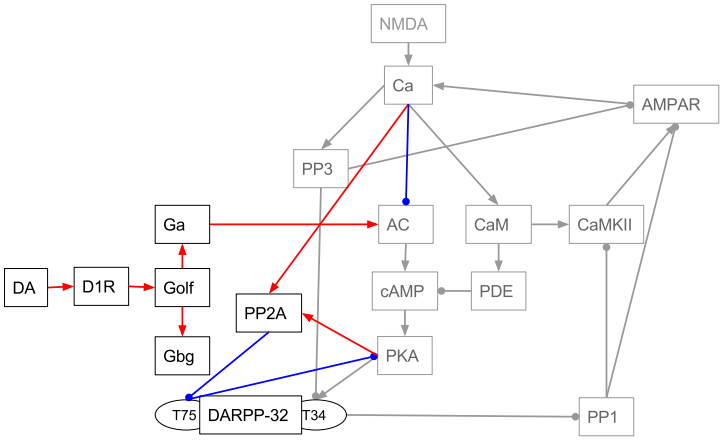


Fig 2. Incomplete block diagrams comparing essential elements of striatal biochemical circuit. Greyed nodes and edges denote shared elements with hippocampal models. See Fig 1 for explanation.

protein synthesis that allows, at other synapses, weak stimuli to cause plasticity.

Models of striatal synaptic signalling pathways

The striatum integrates multiple inputs to the basal ganglia, such as glutamatergic excitatory afferents from the cortex and dopaminergic inputs from the midbrain [102]. Around 95% of striatal cells are MSPNs, in which signalling cascades activated simultaneously by glutamatergic and dopaminergic stimuli is a necessary condition for the LTP that underlies reinforcement learning [103]. Models of striatal MSPNs share some pathways with hippocampal synapses and include striatum-specific proteins.

Multistate DARPP-32 An abundantly expressed protein in MSPNs is phosphatase 1 regulatory subunit 1B (PPP1R1B), known as dopamine- and cAMP-regulated neuronal phosphoprotein with molecular weight 32 kDa (DARPP-32). As a homologue of I1, it has the same major role of PP1 inhibition. It is a hub protein that is regulated by multiple neurotransmitters and phosphorylation sites. There are at least 8 modification sites known in the DARPP-32 amino acid sequence, and 4 of them are known to have a regulatory impact on DARPP-32 [104]. The threonine sites (Thr34 and Thr75, as positioned on the rat protein sequence) have a major regulatory role in signal processing. Thr34 inhibits PP1 and is phosphorylated by PKA, which Thr75 inhibits. The serine sites (Ser137, Ser102, as positioned on the rat protein sequence) regulate Thr34 positively. Ser137 inhibits dephosphorylation of Thr34 on Ca^{2+} stimulation and Ser102 enhances phosphorylation of Thr34. A number of models of dopamine (DA) and Ca^{2+} signal integration have included only Thr34 and Thr75 as major switching factors between LTP and LTD [17, 18, 29, 105]. A few models incorporate all four phosphorylation sites [19, 106].

Glutamatergic and dopaminergic signal integration Lindskog et al. [105] created an ODE model of interacting cascades activated by DA and Glutamate (Glu) signals stimulating dopamine receptor D1 (DRD1) and Ca^{2+} influx through NMDAR, respectively. The glutamatergic signalling cascade shares the general network structure of the CaMKII circuit with hippocampal models (Fig 2), with a few major differences.

Firstly, the inhibition of PP1 does not occur via I1 but rather via DARPP-32 phosphorylated at Thr34. Secondly, as the DRD1 is a G-protein-coupled receptor (GPCR), DA input adds to the network G-protein activation events. On DA stimulation, $G_{\alpha\beta\gamma}$ dissociates into $G_{\alpha,\mathrm{olf}}$ and $G_{\beta\gamma}$ subunits. Subsequently, $G_{\alpha,\mathrm{olf}}$ binds to AC and ATP, synthesising cAMP. The last event, which results in activation of PKA and the cascade inhibiting PP1, is shared by both hippocampal and striatal models. However, in contrast to hippocampal models, in Lindskog's model [105], Ca^{2+} inhibits AC, leaving its activation to DA input. Furthermore, Ca^{2+}-activated PP3 dephosphorylates Thr34 counteracting the DA, but not the Ca^{2+} signal.

In the model Thr34 is both activated and inhibited by a Ca^{2+} feedforward signal, which is conveyed by the PKA–PP2–Thr75 double negative feedback loop. PP2 dephosphorylates Thr75 but its action is enhanced by Ca^{2+} and PKA. The model showed that the loop does not exclusively reinforce PKA pathway stimulated by DA but instead acts as a competitive inhibitor for PKA.

The detailed model of Nakano et al. [15] demonstrated that the loop can have a major role in LTP induction. They extended the network upstream of DARPP-32 and added AMPAR phosphorylation and trafficking as a direct readout of plasticity. Their model required activation of both CaMKII and PKA to reach striatal LTP. They also included the downstream pathway of mGluR activation that represented mainly the bi-directional effect of Ca^{2+} on IP_3 receptor located at the endoplasmic reticulum.

STEP-mediated crosstalk between glutamatergic and dopaminergic signalling cascades Gutierrez-Arenas et al. [18] developed a signalling model of two main signalling pathways activated by DA and Glu inputs in MSPNs: AC–cAMP–PKA and NMDAR–Ca^{2+}–Ras. The AC-pathway was built on the model of Lindskog [105] by adding a NMDAR-cascade, in which the dissociated $G_{\beta\gamma}$ subunits activate Fyn which phosphorylates a NMDAR subunit, thus enhancing the Ca^{2+} influx. Ca^{2+} activates the MAPK pathway phosphorylating mitogen-activated protein kinase 1, also known as ERK2 (MAPK1) at two sites. In striatal plasticity, MAPK1 activation is known to require both DRD1 and NMDAR stimulation, as shown by the negative impact on MAPK1 phosphorylation in the DARPP-32-knockout mouse model [107]. DRD1 activation by the DA-signal also enhanced the Ca^{2+} current through NMDAR, e.g. by the phosphorylation of NMDAR by activated PKA. This particular reaction network was chosen to allow for examination of various scenarios that could explain the results of behavioural experiments showing distinctive segregation of behaviours of two animal types representing $G_{\alpha,\mathrm{olf}}$-deficiency and DRD1-deficiency. The former exhibited disruption of phosphorylation of the GluR1 subunit of AMPAR and the latter disrupted phosphorylation of MAPK1 after acute psychostimulant administration. This effect was present despite known crosstalks between two cascades mediated by striatal enriched tyrosine phosphatase (STEP), which could balance the sensitivity in both pathways. The model reproduced the segregation with an assumption that there are two DRD1/G_{olf} signalling compartments for each pathway distributed from common pools of DRD1 and G_{olf}. These compartments differ in DRD1 and G_{olf} distribution determined by the opposite affinity strengths for these molecules in each compartment. These settings resulted in a competition between the two compartments for G_{olf}/DRD1 resources, giving a 'winning hand' to the one with a stronger affinity to a given molecule.

Interactions between G-protein-coupled receptors DRD1 is a subfamily of dopamine receptors and one of multiple types of GPCRs expressed in MSPNs, including serotonin (5-HT_{2C} receptor [108]), noradrenaline (α_2-adrenoceptor, β_1-adrenoceptor [109], acetylcholine (muscarinic M4 receptor; M4R), adenosine (A2a receptors; A2aR) and dopamine receptors of D_2-like family. The last three, alongside

DRD1, were modelled by Nair et al. [17], who simulated the reward prediction error (defined as the difference between the received and expected reward). They modelled two types of MSPNs, expressing either DRD1 and M4R (striatonigral projections) or DRD2 and A2aR (striatopallidal projections). These two types of neurons process DA-signals in two opposing manners by stimulating (DRD1-expressing) or inhibiting (DRD2-expressing) the signalling cascade resulting in phosphorylation of DARPP-32 at Thr34. In both models neuromodulators interact through $G_{i/o}$ and G_{olf} signalling, inhibiting and activating AC5 respectively. Also in both models, AC5 is inhibited by $G_{i/o}$ at the basal state. In the DRD1-expressing neurons, $G_{i/o}$ is coupled with the M4R–tonic ACh signal; and in the DRD2-type of neurons with the DRD2–tonic DA signal. In DRD1-neurons, the high PKA activation level was achieved with a simultaneous DA-peak and ACh-dip. These neurotransmitter signals realise an AND-gate, sensitive but noise-prone to a positive reward. In DRD2-neurons it is the DA-dip that increases the PKA activation, even without Adn signal. This suggests that in this type of neurons the cAMP–PKA cascade mainly detects reward omission.

Spatial specificity in synaptic plasticity The model of Oliveira et al. [29] studied the mechanisms of spatial restriction of PKA activation by A-kinase anchoring protein (AKAP). The problem required a multi-compartmental stochastic reaction-diffusion approach. To evaluate distinct functions of anchoring, the experimental protocol consisted of four spatial variations in localisation of AC and PKA, either locating them in the spine head or at dendritic submembrane area. The signalling network was adopted from Lindskog [105] and the stimulating signal was either dopamine alone, corresponding to the reward response, or the combined DA and Ca^{2+} influx used for LTP protocols. The results showed that for the induction of LTP the colocalisation of PKA near the source of cAMP is more important than its colocalisation near its target substrates (e.g. DARPP-32, PP2, PDE).

Kim et al. [31] used the NeuroRD algorithm to model 19 molecules in the postsynaptic signalling pathways of the dendrites of striatal MSPNs with multiple spines. The model investigated the hypothesis that temporal patterns, linked to Ca^{2+}, determine LTP or LTD induction, via PKC or endocannabinoid 2-arachidonoyl-glycerol (2AG) production respectively. The ratio between the number of activated PKC and 2AG molecules was used as an indicator of the direction of plasticity. It describes G_q-coupled pathways, the temporal pattern of Ca^{2+} stimulation and $G_{\alpha,q}$ activation. In the simulations LTP was specific to spines, whereas LTD was more diffuse. This suggested that spatiotemporal control of striatal information processing uses G_q-coupled pathways for decision-making.

Cerebellar synaptic models

Despite the historical importance of cerebellar granule cell to Purkinje cell plasticity, at least 9 types of synaptic and non-synaptic plasticity are known [110]. The classical LTD at cerebellar granule cell to Purkinje cell synapses occurs when there is simultaneous climbing fibre and granule cell (parallel fibre) firing. At the heart of the model of Kuroda et al. [111] is the MAPK positive feedback loop found in hippocampal and striatal models [18, 80], which here comprises Raf–MAP2K–MAPK-PLA2–AA–PKC. Parallel fibre activity both activates and inhibits the loop. Parallel fibre glutamatergic input to AMPARs causes Na^+ influx, which triggers the Na^+/Ca^{2+} exchanger causing Ca^{2+} influx which, in turn, activates PKC and PLA2. PKC is also activated via mGluR and AMPARs also activates Lyn tyrosine kinase directly, which activates Raf in the MAPK loop. Parallel fibre input also releases NO, which, via the guanylate cyclase–cGMP–PKG pathway, activates PP2, which inhibits MAP2K. Climbing fibre

inputs also activate the MAPK link via Ca^{2+}, and via Raf which is activated corticotropin releasing hormone receptors (CRHR) activated by corticotropin releasing factor. When the loop is active, activated PKC phosphorylates AMPARs, but in contrast to hippocampal models phosphorylated AMPARs are internalised, leading to LTD.

Antunes and DeSchutter [112] model LTD in cerebellar granule cell to Purkinje cell synapses in the cerebellum using Gillespie's SSA, as implemented in the STEPS simulator. The model includes a version of the PKC-MAPK circuit (Fig 2), but with an undetermined "Raf-activator" between PKC and Raf. This Raf-activator could be Ras itself or indirect activation of Ras via complex Src/Proline-Rich Tyrosine Kinase 2 (PYK2). PP5 tonically inhibits Raf and MKP (DUSP) inhibits MAPK. Activated PKC promotes endocytosis of AMPARs, thus causing LTD. The stochastic nature of the model leads to LTD being stochastic and binary at individual synapses, but over the ensemble of synapses this results in a graded relationship with the magnitude of the activating Ca^{2+} signal. Increasing the number of molecules makes the system less stochastic, and makes the resulting macroscopic signal less graded.

Antunes et al. [42] extend this model by incorporating CaMKII and PP3 to implement LTP. They use the rule-based BioNetGen system to generate stochastic reactions that are simulated using Gillespie's SSA. In contrast to hippocampal models, calcineurin promotes LTP by preventing endocytosis of AMPARs. RKIP is also incorporated as an additional activator of Raf.

Summary

In summary, the development of biophysical models of synaptic plasticity has been propelled by: (1) hypothesis-driven physiological and molecular biological discoveries; (2) the need to formalise informally expressed hypotheses; (3) the intrinsic fascination and intellectual challenge of complex biomolecules such as CaMKII; and (4) increasing compute power, which makes it practical to model stochastic and spatial aspects of synaptic signalling cascades. Challenges in the field have included dealing with combinatorial complexity and finding appropriate sets of parameters. Recent computational modelling methods, such as agent-based and particle based simulation, address the problem of computational complexity. Depspite being an active field of research, the perennial problem of inferring parameter values remains more intractable.

Analysis of proteins in synaptic models

Computational models of synaptic plasticity are important tools for understanding synaptic and neural function. When they include molecular entities and phenomena they can also be used to study dysfunction, and potentially model pharmacological interventions. Clearly the coverage of synaptic molecules found in the existing 'model space' is going to be very incomplete given the intense amount of effort required to develop each model but here we sought to explore systematically molecular coverage to identify significant gaps that might offer new opportunities.

Computational models contain a diverse cast of players, including proteins, second messengers, reporters, ions and others. Models vary in how precisely they specify proteins; for example Bhalla and Iyengar [80] specify AC1, AC2 and AC8, whereas Castellani et al. [82] and Oliveira et al. [28] specify AC, which could, in principle, map to any of the adenylate cyclases expressed in the synapse. This presents a problem when mapping models to molecular identifiers, which we addressed by developing a mapping from what we refer to as model "entities" to gene families. For example a protein such as Calmodulin 1 can be mapped onto a single gene (*CALM1*), but a family of proteins

such as metabotropic glutamate receptors maps onto more than one gene (*GRM1–GRM8*). By definition, second messengers or ions do not map onto gene symbols.

The concept of entities allows each model's constituents to be catalogued faithfully and then mapped onto identifiers according to the steps shown in Fig 3: (1) select models to analyse; (2) determine all entities (e.g. proteins, protein multimers or families, ions and second messengers) that are contained in each model; (3) map these entities onto gene identifiers and higher level families; and (4) use the lists of entities in each model and the mappings to undertake comparative analyses. These analyses include: comparison of modelled proteins with pre- and postsynaptic proteomic datasets; identification of properties of modelled genes, in particular cellular pathways, gene ontology terms and disease; and comparison of models with each other.

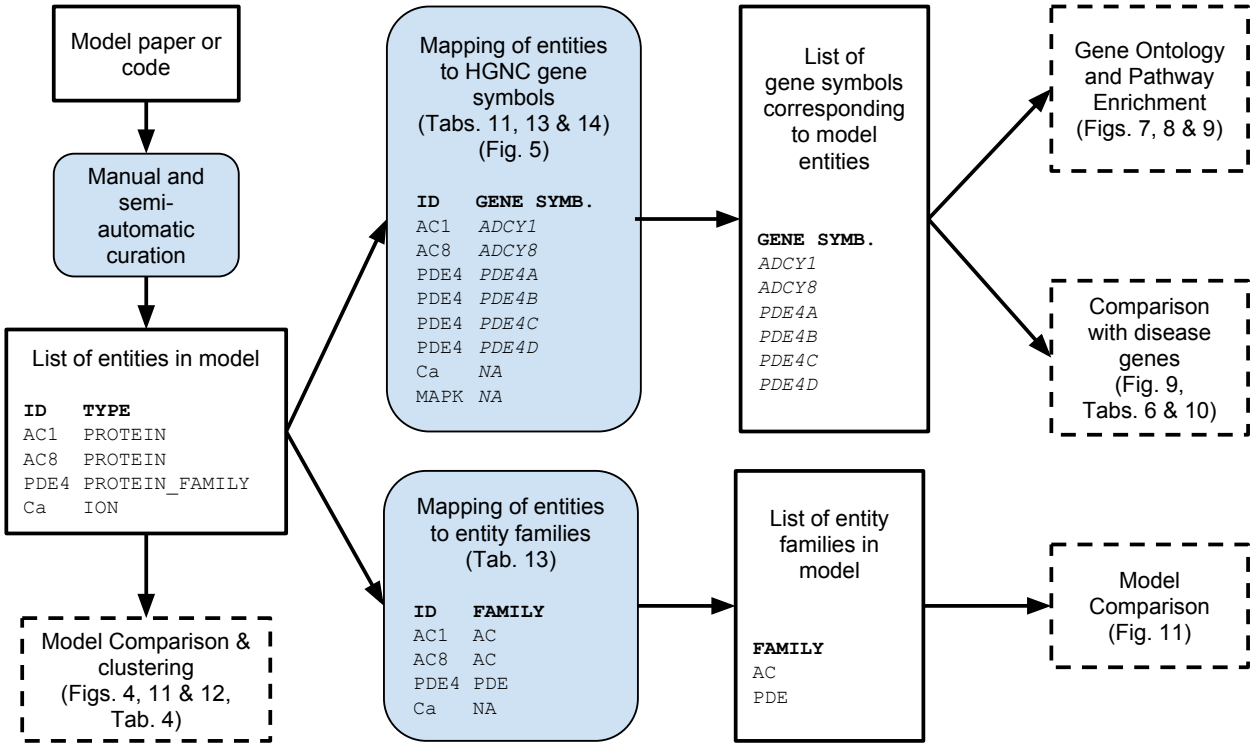


Fig 3. Overview of the modelling paper analysis process. Sets of data are shown in boxes with black rectangular borders. Processes are shown in boxes with blue backgrounds and curved corners. Final analyses are shown in boxes with dashed borders. "ID" refers to the modelled entity. Boldface type refers to column headers.

Selection of models

We selected a number of published computational, biophysical models of synaptic plasticity or related pathways (Table 3). Models that we regarded as phenomenological or descriptive, i.e. models describing a function with no explicit reference to an underlying mechanism, were excluded. For example, models of spike-timing dependent synaptic plasticity are phenomenological, since they contain an empirical function that maps spike times onto changes in plasticity with no reference to proteins.

The process of identifying the model constituents can be time-consuming, especially when machine-readable descriptions are not available. In order to address our questions regarding the molecular coverage of synaptic models, it sufficed to select a set of models that we were reasonably confident gave good genetic coverage, rather than to identify entities in every model. We assessed molecular coverage of pre-2010 models from the tables in Manninen et al. [9] and we screened models published between 2010 and December 31st 2015.

Sources of models

A number of the models we selected are written in standardised modelling languages and hosted in large scale repositories such as ModelDB [113], BioModels [114], DOQCS [115] and the CellML repository [116]. ModelDB is a curated database of computational neuroscience models at the molecular and electrophysiological levels, written in a number of languages. BioModels hosts models which focus on biochemical and cellular systems at the physiological and biochemical levels, unrestricted by the biological subject [114, 117]. In the curated branch of BioModels, models have to be annotated according to the minimal information requested in the annotation of biochemical models (MIRIAM) standard [118], thus meaning that model constituents are mapped to external identifiers. CellML is both a model format and a repository. The repository hosts a wide range of biological models, which have documentation pages generated from the meta-data supplied by model authors. DOQCS (Database of Quantitative Cell Signalling) is a database tailored for storing chemical kinetics and reaction level information [115]. The chemical-level description of each model corresponds to the GENESIS/Kinetikit simulator and reflects reaction diagrams or ODE equations.

Table 2 summarises the numbers of models we analysed that are stored in repositories and other locations, and the format of the model descriptions. Three of the 7 models deposited in the BioModels database were curated to MIRIAM standards. Around half of all catalogued models (14) had non-machine readable descriptions. Models in this group are often difficult to explore and extract information proves challenging. There were 18 machine-readable models available from publication attachments, on institute or lab servers and the four public modelling databases; some models are deposited in more than one database. With two exceptions models were not duplicated in ModelDB and BioModels; the Bhalla and Iyengar [80] model was present in all four public modelling databases, and the Nakano et al. [15] model was found in ModelDB and BioModels. We did not test the functionality or reproducibility of models; only the availability and relative ease of exploration was examined.

Features of models

We extracted a number of features from each model to highlight their similarities and differences (see Table 3). To quantify the model size, we counted the number of entities that appear in the model. We also extracted information on numbers of dynamic variables per compartment ("Vars/comp."). Variables are values describing quantities that change in the model. A compartment is defined as a spatial subsection within the

Table 2. Overview of locations of models and their formats.

Type	Location		Format	Fraction
non-machine-readable	attached to publication or within publication content		appendix, doc, pdf, excel or descriptions, reaction diagrams, equations	14/30
machine-readable	attached to publication		software-specific	3/30
	institutes, labs servers			3/30
	public modelling databases	ModelDB	any (software-specific): NEURON, Python, C, C++, GENESIS, Java, Matlab, XPP, etc.	8/30
		BioModels	all (automatically translated): SBML, CellML, VCML, XPP, SciLab, Octave, BioPAX	7/30
		CellML	CellML	1/30
		DOQCS	GENESIS	2/30

Fractions refer to the number of models in the category relative to the total of annotated models. Each machine-readable model can be part of several categories. See text for details.

model. Since the number of compartments varies with the fineness of the spatial mesh used, the number of variables scales with the number of compartments, but the number of variables per compartment will be a constant, independent of the spatial discretisation used to simulate the model. To provide a measure of model complexity, we used the ratio of thethe number of variables per compartment and the number of entities ("Vars./Comp./Entities", Table 3).

For example, in a model of calcium binding to a buffer in a single compartment, there are two entities: calcium (an ion) and the buffer (a protein). There are three variables, namely the concentrations of free calcium, free buffer and calcium-buffer complex. To model diffusion of calcium, buffer and calcium-buffer complex, space could be divided into 100 compartments. The number of variables would then be 300, but the number of variables per compartment would be 3. There would still only be two entities in this model – calcium and the buffer – and the variables per compartment per entity ratio would be 1.5.

A high ratio of variables per compartment to entities reflects a detailed description of a small pathway. For example the model of Byrne et al. [58] – whose stochastic model describes binding of calcium, calmodulin and CaMKII – has 82 variables per compartment and 3 entities, making a ratio of 27.3. The 82 variables correspond to the combinations of calcium bound to the N and C lobes of calmodulin and whether or not these complexes are bound to CaMKII. Dealing with this complexity in the simulation is achieved by using an agent-based Gillespie method (Section "Non-spatial models" in "Biophysical models of synaptic plasticity"). Agent-based simulation also allows the more extreme example of Zeng and Holmes [27], whose model of the Ca^{2+}-CaM-CaMKII-PP3 pathway (with calbindin and neurogranin; 6 entities in total) has 14,296,081 possible complexes (i.e. variables), making a ratio of 2,382,680 variables per compartment per entity. At the other end of the spectrum, a low variable to entity ratio indicates larger pathways with each interaction modelled in less detail. For example, the ODE-based model of Bhalla and Iyengar [80], with 44 entities and approximately 100 variables per compartment, has a ratio of 2.3 variables per compartment per entity.

In Table 3 we also indicate the region or cell type the model applies to. Hippocampal CA1 cells are most frequently modelled, followed by striatal MSPNs and cerebellar Purkinje neurons. In some models the location is not specified.

Table 3. Summary of models.

Paper	Vars./comp.	Entities	Vars./comp./ Entities	Region
Antunes and De Schutter (2012) [112]	103	19	5.4	Cereb. Purk.
Antunes et al. (2016) [42]		17		Cereb. Purk.
Bhalla and Iyengar (1999) [80]	100	42	2.4	Hipp. CA1 Pyr.
Byrne et al. (2009) [58]	82	3	27.3	Hipp. CA1 Pyr.
Castellani et al. (2001) [90]	36	5	7.2	Cortex**
Castellani et al. (2005) [82]	33	13	2.5	Ex. glut. syn.**
Graupner and Brunel (2007) [22]	16	5	3.2	Hipp. CA1 Pyr.
Gutierrez-Arenas et al. (2014) [18]	188	34	5.5	Striatal MSPN, D1R expressing
Hernjak et al. (2005) [26]	9	5	1.8	Cereb. Purk.
Khan et al. (2011) [32]	12	1	12.0	Hipp. CA1 Pyr.
Kim et al. (2010) [21]	54	18	3.0	Hipp. CA1 Pyr.
Kim et al. (2011) [30]	16	17	1.0	Hipp. CA1 Pyr.
Kim et al. (2013) [31]	10	18	0.6	Striatal MSPN, mGluR1 expressing
Kötter (1994) [119]		12		striatal MSPN
Kuroda et al. (2001) [111]		20		Cereb. Purk.
Li et al. (2012) [33]	95	8	11.9	Generic excitatory spine
Mattioni and Le Novère (2013) [34]	13	9	1.4	Striatal MSPN
Miller et al. (2005) [49]	58	4	14.5	**
Nair et al. (2015) [17]	80	16	5.0	Striatal MSPN, D1R and D2R expressing*
Nakano et al. (2010) [15]	189	28	6.8	Striatal MSPN, D1R expressing
Oliveira et al. (2010) [28]	31	9	3.4	HEK293 cells
Oliveira et al. (2012) [29]	113	28	4.0	Stratial MSPN
Pepke et al. (2010) [20]	156	3	52.0	**
Qi et al. (2010) [19]	115	13	8.8	Stratial MSPN
Smolen et al. (2006) [100]	23	9	2.6	Hipp. CA1 Pyr.
Smolen et al. (2012) [16]	14	6	2.4	Hipp. CA1 Pyr.
Sorokina et al. (2011) [23]	1,000,000	55	18,181.8	Ext. glut. syn.
Stefan et al. (2008) [59]	49	3	16.3	**
Zeng and Holmes (2010) [27]	14,296,081	6	2,382,680.2	Hipp. DG
Zhabotinsky et al. (2006) [83]	58	11	5.3	Hipp. CA1 Pyr.

"Paper" refers to the analysed model. "Vars/comp." is the number of molecular variables per compartment, a measure of the complexity of the model; this was not assessed for all papers. "Entities" is the number of entities in the model, and "Vars./Enties" is the ratio between the number of variables per compartment and the number of entities. This roughly corresponds to the level of detail of the model. "Region" refers to the brain region or cell type where the model is situated (** – no cell specified). Abbreviation: Cereb. Purk., cerebellar Purkinje cell; Ex. glut. syn., excitatory glutamatergic synapse; Hipp. CA1 Pyr., hippocampal CA1 pyramidal cells; Hipp. DG, hippocampal dentate gyrus cell; MSPN, medium spiny projection neuron; * – denotes that there is more than one model presented in a study and numbers in this table refer to the one with the larger number of "Entities".

Table 4. Frequency of entity types found in models.

Type	Frequency	Examples
Ion	2	Magnesium, Calcium
Neurotransmitter	5	Adenosine, Dopamine
Others	2	ATP and PIP2, intermediates in the IP3/DAG pathway
Protein	95	Neurogranin
Protein family	52	calmodulin, which may correspond to one of calmodulin-1, calmodulin-2 or calmodulin-3
Protein multimer	8	AMPA receptor, which comprises a tetramer of GluR1, GluR2, GluR3 and GluR4 proteins.
Reporter	1	AKAR3
Second messenger	8	GTP (Guanosine triphosphate) or cAMP (cyclic AMP).
Total	173	

Identifying entities in models

To identify the entities in each model, the publication describing the model and, if available, an electronic description of the model were examined by one of the authors. For each entity, we recorded the name used in the model publication and our standard entity identifier. Models do not always specify the entities involved precisely. We discussed ambiguous cases together and erred on the side of not imputing the identity of a protein; for example a "Plasticity related protein" [16] was not mapped to an entity identifier.

We identified 178 distinct entities across the 30 catalogued models (see S1 Table for full list). As well as an identifier, each entity has a long name and a type which can be one of: "ion", "neurotransmitter", "others", "protein", "protein family", "protein multimer", "reporter" or "second messenger". Table 4 shows how many of each type of entity were identified, and gives examples. The most frequent entity type is "protein", followed by "protein family" and then "protein multimer".

The rationale for having three protein types – "proteins", "protein families" and "protein multimers" – was to allow us to record as precisely as possible what was meant in each computational model. A "protein" is a specific protein e.g. neurogranin, encoded by a specific gene (*NRGN*), so it is unambiguous as to which gene is implied by the model. The same gene may produce multiple isoforms due to gene duplicates or alternate splicing. For example *PRKCZ* produces two isoforms, PKCζ and PKMζ [120]. A "protein multimer" is a multiprotein complex, e.g. an AMPA receptor, which comprises a tetramer of a selection of GluR1, GluR2, GluR3 and GluR4 proteins. In this example, if the model only specified "AMPAR" there would be ambiguity about which of the GluR1–4 subunits are implied by the model. Coding AMPAR as a "protein multimer" allows this ambiguity to be recorded and resolved as desired. A "protein family" is a protein from a family of proteins, e.g. calmodulin, which may correspond to one of calmodulin-1, calmodulin-2 or calmodulin-3. Again, it is not clear which protein is implied by the model, though later we will use information about the synaptic proteome to narrow down the possibilities. "*AKAR3*" is the only entity that was classified as a reporter [17]. The FLIM-AKAR reporter was included in the model to reflect the experimental setup where it is used to measure PKA dynamics. "Ions", "neurotransmitters" and "second messengers" were assigned to individual classes. They are not proteins, but carry out crucial functions in the cell.

ATP and PIP2, both intermediates in the IP3/DAG pathway were classified as "other". ATP itself can produce a second messenger and is often referred to as a precursor or "coenzyme". Similarly, PIP2 is frequently acting as a precursor of a second messenger [31].

The full catalogue of all model entities for all models is shown in matrix form in Fig 4. The models are ordered according to hierarchical clustering (Ward's 2D method, as implemented in R's `hclust` function with the `Ward.2D` method). This catalogue is the basis for the rest of the analysis.

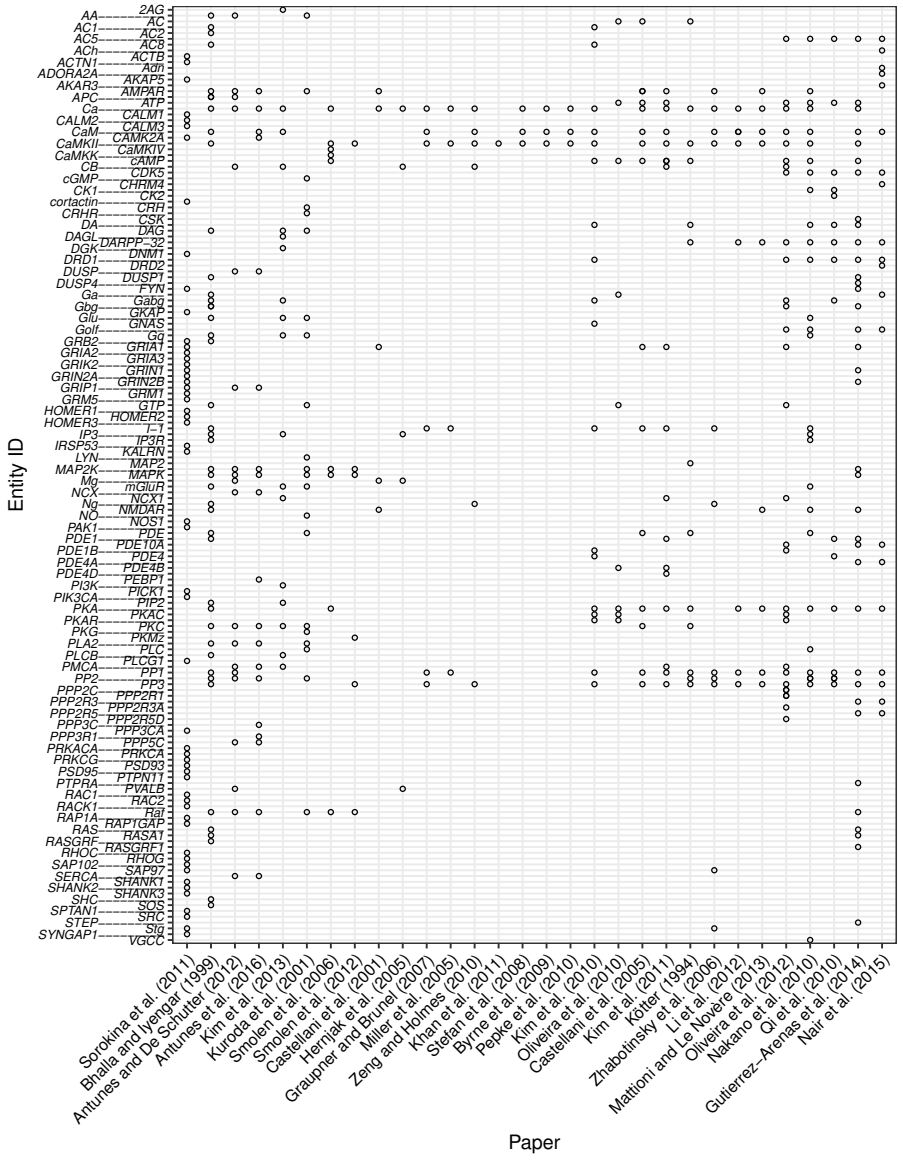


Fig 4. Matrix of entities in models. The occurrence of an entity in a model is indicated by open circles. Entity IDs are staggered for readability.

Mapping entities to gene identifiers

In order to compare synaptic models with the synaptic proteome, we needed to map each protein entity onto the proteins to which it might correspond. The construction of this mapping is shown in Fig 5. Based on common practice in bioinformatics we decided to use HUGO Gene Nomenclature Committee (HGNC) gene symbols and NCBI Entrez Gene IDs to identify proteins/genes. The one-to-one mapping from HGNC gene symbols to NCBI human Entrez Gene IDs [121] allowed this approach.

As presented in Fig 5, entities of type "protein" were mapped directly to HGNC gene symbols. Entities classified as "protein family" and "protein multimer" required an intermediate mapping step. We searched for ontologies that could be used to identify as many of these entities as possible and map them to HGNC gene symbols. After thorough analysis of available bioinformatic resources (see Methods) we decided to use HGNC gene families to map entities of type "protein family" and "protein multimer" to genes. For each such entity, we tried to identify a corresponding HGNC gene family, and used manual NCBI mapping (see Methods) to check if the genes contained in this family seemed likely to be what was meant in the models. For example, we mapped the entity "Dopamine receptors" (DRD) to the HGNC family "Dopamine receptors", which contains the genes *DRD1*, *DRD2*, *DRD3*, *DRD4* and *DRD5*. Since this seemed a reasonable set, we accepted the mapping.

For some entities no one HGNC family gave a reasonable set of proteins, but the intersection between two or more families did. For example the genes corresponding to SHANK, by which we mean the family of proteins encoded by *SHANK1*, *SHANK2* and *SHANK3*, may be selected from the gene families list by choosing all genes that are in the "Ankyrin repeat domain containing" (ANKRD) and "PDZ domain containing" (PDZ) gene families. When we could not find a corresponding HGNC family or a combination of HGNC families, we constructed our own mapping (see Methods). Since "ions", "neurotransmitters", "others", "reporters" and "second messengers" are not proteins, we excluded them from the mapping to gene names.

Once gene families corresponding to 61 "protein families" and "protein multimers" were identified we could map each family or multimer onto a set of genes (S3 Table and S4 Table). 331 unique HGNC gene symbols were identified based on protein families and multimers. The union of this set of symbols with the 96 genes mapped directly from type "protein" in the "full set of HGNC gene symbols in models" dataset. It contains a total of 386 HGNC gene symbols. A number of "protein families" mapped onto the same genes; for example the families PDE and PDE1 both contain *PDE1A* and *PDE1B*.

Comparison with proteomic data

HGNC families are general gene classes and do not contain information about tissue specificity or expression patterns. To identify proteins found in the synapse, we used a meta-analysis of published proteomic datasets of the presynapse, postsynapse and synaptosome that we are preparing for another publication. The individual references, as of July 2017, can be found in S2 Table.

The synaptosome is the largest data subset and extracted from brain homogenate. The term synaptosome refer to the complete presynaptic terminal including mitochondria, synaptic vesicles and the postsynaptic membrane together with the PSD [122, 123]. The PSD is a tightly connected, dense region of the postsynaptic membrane which hosts a number of different receptors and regulatory units. The presynapse and postsynapse are subsets of the synaptosome, and can be separated through experimental steps.

The union of these three datasets, which we refer to as the "synaptic proteome", comprises 6,706 genes and is based on data obtained from 37 publications and 39

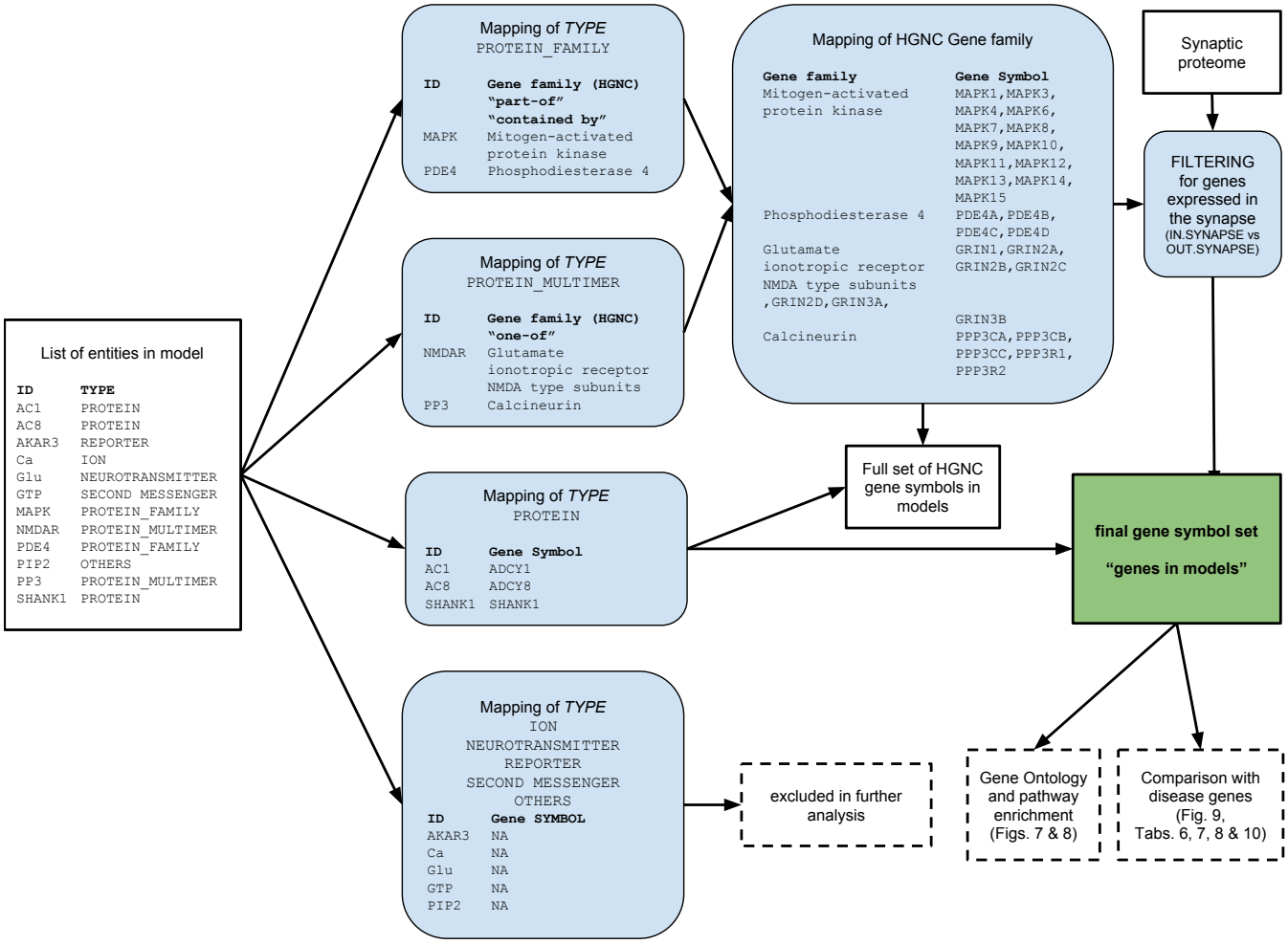


Fig 5. Overview of entity to Gene Symbol mapping process. Sets of data are shown in boxes with black rectangular borders. Mappings are shown in boxes with blue backgrounds and curved corners. Dashed lines indicate additional information, and the key outcome is highlighted in a box with green background. Bold font refers to column headers.

datasets (data as of July 2017). The extracted proteome was used to filter the "full set of HGNC Gene symbols in models" (see Fig 5 and "Identifying entities in models"). We found that every "protein family" (S3 Table) and "protein multimer" (S4 Table) in our list contains at least one gene overlapping with the synaptic proteome. Genes not expressed in the synapse ("OUT SYNAPSE" in S3 Table and S4 Table) were excluded from further analysis. This filtering step reduces the 331 genes in families to 239 HGNC gene symbols. Together with directly mapped proteins this leaves us with 294 unique HGNC gene symbols describing all mapped genes in models, where families and multimers were screened for the presence in the synapse. From now on we refer to this gene set as "genes in models" (see green box, Fig 5).

The overlap between the final set of "genes in models" and the synaptic proteome, as well as its subsets (presynaptic, postsynaptic, and synaptosome) is visualised in the

Venn diagram in Fig 6. It can be seen that 46% of "genes in models" (135 genes) are found in all three synaptic proteome datasets. Significantly lower numbers are expressed in individual sub-datasets. These are 3, 14 and 21 genes for the presynapse, postsynapse and synaptosome respectively (representing 1.0%, 4.7% and 7.1% of genes in models). When disregarding "genes in models" present in the intersection of all three datasets, more modelled genes are found in the postsynapse or synaptosome (143 genes) than the presynapse or synaptosome (27 genes). Thus, postsynaptic genes appear to be the most highly modelled subset. However, relative to the total size of the respective proteomes, only 5.1% of postsynaptic genes (258 "genes in models" out of 5,053 postsynaptic genes) versus 7.6% of presynaptic genes (142 "genes in models" out of 1,867 presynaptic genes) are represented in the models.

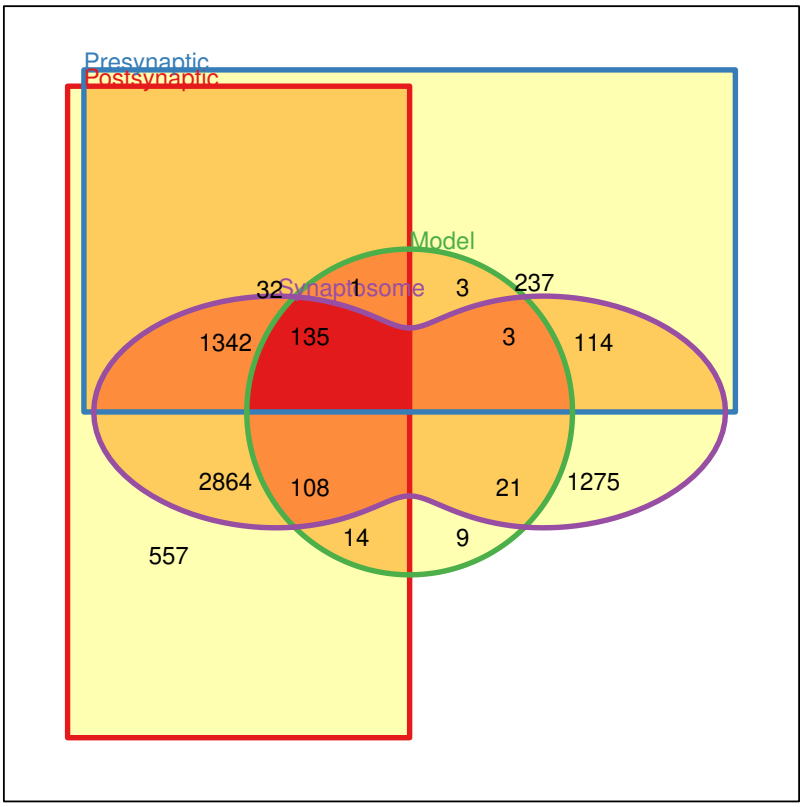


Fig 6. Relationships between the sets of genes in postsynaptic, presynaptic, synaptosome datasets and the sets of genes possibly present in models. Postsynaptic genes in red, presynaptic in blue, the synaptosome in purple and genes in models in green. Numbers refer to the number of genes in each subset and shading shows how many sets a region belongs to (white – none; red – all four). It can be seen that the number of genes in the proteome but not included in models is an order of magnitude bigger than the number of proteins included in models and the proteomic datasets. There are only 9 genes (listed in Table 5) found in models and none of the proteomic datasets.

Nine modelled genes, all of type "protein" are not present in the synaptic proteome

datasets (see lower right of the circle in Fig 6). Further investigation shows evidence for all of them being expressed in the synapse (Table 5), so these 9 genes remained in the set of "genes in models". These cases illustrate how proteomic datasets still seem to be slightly incomplete.

Table 5. Proteins in models and not to be found in synaptic datasets.

Entity ID	Gene	Reason for inclusion
ADORA2A	*ADORA2A*	Adenosine A2a receptors (A2aR) are expressed with D2R receptors [17]
CALM2	*CALM2*	Unpublished dataset
CHRM4	*CHRM4*	Muscarinic cholinergic receptor shown to be expressed in gonadotropin releasing hormone neurons [124]
CRH	*CRH*	Corticotropin-releasing factor, regulating the release of adrenocorticotropin in synapses [125]
DRD1	*DRD1*	D1 subtype of the G-protein coupled dopamine receptor - the most abundant in the central nervous system. [126] confirms the presence in neurons.
DRD2	*DRD2*	D2 subtype of the G-protein coupled dopamine receptor. [126] confirms the presence in neurons.
DUSP1	*DUSP1*	Model specifies that DUSP1 feedback loop occurs in the dendritic shaft, the soma and the nucleus [18]
I-1	*PPP1R1A*	Unpublished dataset
PPP2R3A	*PPP2R3A*	Preliminary studies suggest PPP2R3A is present in both cytoplasm and nucleus of cells in the striatum [127]. PPP2R3A mediates Ca^2-dependent dephosphorylation at Thr-75 of DARPP-32 [127].

Enrichment analysis of modelled genes

After compiling the "genes in models" list, we related it to existing biological knowledge, in the form of gene sets annotated with various biological categories, supplied through a number of databases. Depending on each database's focus, structured, controlled, and descriptive terms are associated to each gene. As an example for this study, we chose to use the following ontologies: Gene Ontology (GO) [128], REACTOME Pathway Database (REACTOME) [129] and Disease Ontology (DO) [130]. Amongst these GO is the largest and most commonly used ontology, classifying genes within domains including Molecular Function, Biological Process and Cellular Compartment. We also used REACTOME, a free and manually curated database in which genes are tagged with terms representing biochemical reactions and pathways they are involved in. A pathway is composed of one or more reactions or reaction-like events, such as binding, complex formation, transport or polymerisation.

To relate "genes in models" to their associated diseases, we used the DO to provide disease classifications. Multiple sources contain gene disease information. We used annotations retrieved from the GeneRif [131], OMIM [132, 133] and Ensemble Variation [134] databases. Based on annotations in the different ontologies we aimed to identify functionalities shared by the "genes in models". The `topONTO` package implemented in `R` [135] was used to undertake enrichment analysis (see Methods).

The results are summarised using word clouds to show significantly enriched terms, based on GO annotations, describing Molecular Functions (Fig 7A) and Biological Processs (Fig 7B) for our "genes in models". It can be seen that a high number of modelled genes are involved in molecular functions such as "G-protein beta/gamma-subunit complex binding", "G-protein beta/gamma-subunit complex

binding", "GTPase activity", "calmodulin binding", "3',5'-cyclic-AMP phosphodiesterase activity", "high voltage-gated calcium channel activity", "signal transducer activity" and "calcium-transporting ATPase activity" amongst others. The most common biological processes are "cellular response to glucagon stimulus", "platelet activation", "calcium ion transmembrane transport", and "activation of protein kinase A activity".

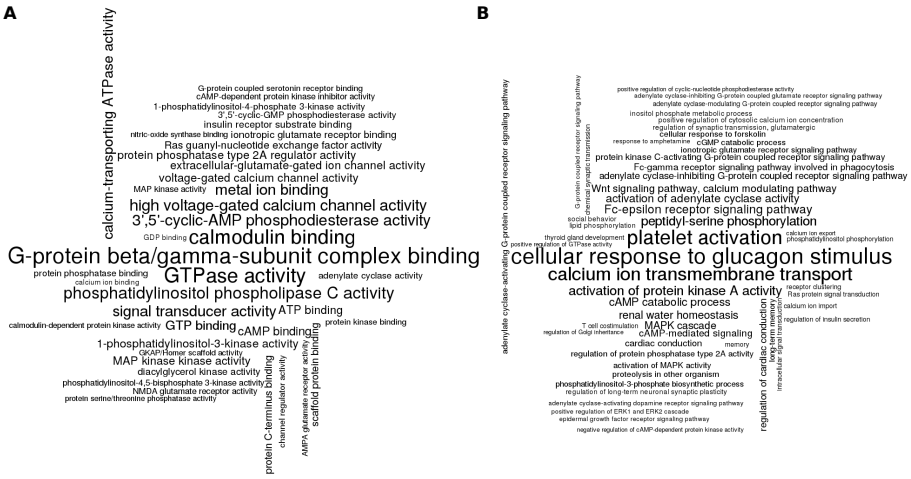


Fig 7. GO enrichment analysis results for "genes in models". A: Molecular Function ontology terms enriched for "genes in models". B: Biological Process ontology terms enriched for "genes in models". The synaptic proteome was used as a background dataset. The list of significant terms was obtained with the Fisher's exact test and the `elim` algorithm, followed by Benjamini and Yekutieli multiple testing correction. The terms shown in clouds scored less than 0.01 p-value after the correction. Font size is proportional to the term significance.

The identified molecular functions show that genes included in annotated models cover key synaptic processes mainly concentrating around energy production as well as synaptic signalling and information transmission. Identified biological processes are slightly more diverse. Fairly generic processes were identified, showing that the set of modelled genes covers these functions in the synapse. More unique processes appear indicating the synapse specific biological processes described by genes in models.

Fig 8 shows results of the REACTOME enrichment analysis that identified "G alpha (s) signalling events", "G alpha (z) signalling events" and "DARPP-32 events" as the top enriched pathways. The first two terms are parallel to each other on the pathway hierarchy and have a common parent term of "GPCR downstream signalling". A comparison of the remaining members of this pathway with the enrichment results shows that they are all significantly enriched in terms of our "genes in models". The identification of signalling pathways highlights a focus of the analysed models indicating the central role of G-protein signalling.

When considering common diseases amongst "genes in models", Fig 9A shows a significant enrichment of "schizophrenia" associated genes in the set of "genes in models", followed by "bipolar disorder", "Huntington's disease" and "Alzheimer's Disease". The order of results is slightly rearranged when considering the whole cell as a background dataset (Fig 9B). For instance, "Alzheimer's Disease" becomes more prominent, showing the second highest significance for enrichment in our dataset of interest. On the other hand, "bipolar disorders" drops down the list to the fifth position

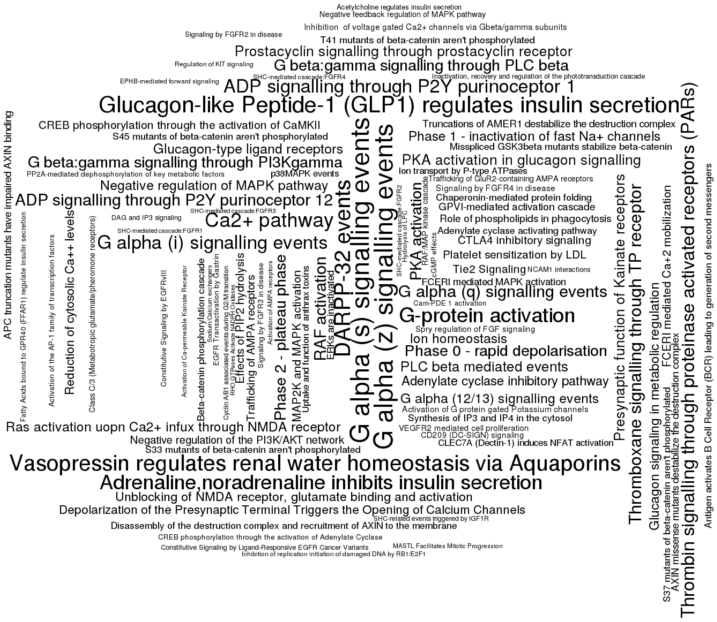


Fig 8. REACTOME enrichment analysis results for "genes in models". The synaptic proteome was used as background dataset. The list of significant terms was obtained with the Fisher's exact test and the `elim` algorithm, followed by Benjamini and Yekutieli multiple testing correction. The terms shown in clouds scored less than 0.01 p-value after the correction.

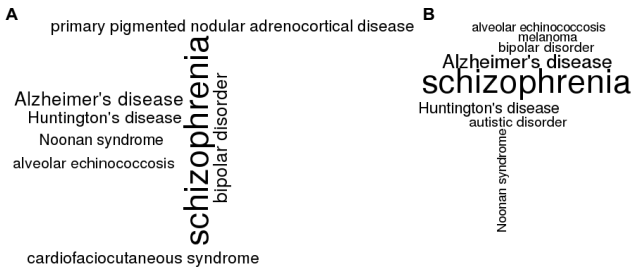


Fig 9. DO enrichment analysis results of "genes in models". Two background datasets were used: synaptic proteome (A) and all human protein coding genes (B). The list of significant terms was obtained with the Fisher's exact test and the `elim` algorithm, followed by Benjamini and Yekutieli multiple testing correction. The terms shown in clouds scored less than 0.01 p-value after the correction.

and "autistic disorder" appears in the results. This shows how different diseases not only affect specific tissues but can affect a larger number of body regions inducing their effect.

Modelled genes and their overlap with disease genes

Based on the preceding enrichment analyses we wanted to test for specific associations of modelled genes with disease. Since synapses play a crucial role in signal transduction and are affected in many neurological diseases, these were addressed in more detail. We

picked seven representative examples of neurological disorders, 6 of which were based on a list published by the Genes 2 Cognition online initiative: *Attention Deficit Hyperactivity Disorder* (ADHD), *Alzheimer's Disease* (AD), *Autism*, *Bipolar Disorder* (BD), *Depression* and *Schizophrenia*. The seventh example was *Parkinson's Disease* (PD), motivated by our research interests. The list is a representative rather than exhaustive sample of diseases affecting synapses, including diseases of mental health, developmental disorders, as well as diseases of anatomical entity, such as neurodegenerative diseases. Table 6 gives the DO identifiers and short descriptions of each disease.

Table 6. Diseases of Interest and short descriptions.

Disease	DOID	Description
Alzheimer's Disease (AD)	DOID:10652	Tauopathy, characterized by memory lapses, emotional instability and progressive loss of mental ability. It results in progressive memory loss, impaired thinking, changes in personality and mood, up to profound decline in cognitive and physical functioning.
Attention Deficit Hyperactivity Disorder (ADHD)	DOID:1094	Specific developmental disorder, characterized by co-existence of attentional problems and hyperactivity.
Autistic Disorder	DOID:12849	An autism spectrum disorder, characterized by symptoms across three symptom domains (communication, social, restricted repetitive interests and behaviors) and delayed language development.
Bipolar Disorder	DOID:3312	A mood disorder that involves alternating periods of mania and depression.
Major Depressive Disorder (MDD)	DOID:1470	An endogenous depression that is characterized by an all-encompassing low mood accompanied by low self-esteem, and by loss of interest or pleasure in normally enjoyable activities.
Parkinson's Disease (PD)	DOID:14330	Synucleinopathy, based on the degeneration of the central nervous system that often impairs motor skills, speech, and other functions.
Schizophrenia	DOID:5419	Psychotic disorder, characterized by a disintegration of thought processes and of emotional responsiveness.

Onto Suite Miner [136] was used to obtain all genes linked to the DO IDs from the databases supplying gene–disease association information (GeneRIF, OMIM and EnsemblVariation). The various databases have different approaches to disease-gene annotations. EnsemblVariation relies on genetic mutations (mostly Single Nucleotide Polymorphisms, SNPs), whereas OMIM and GeneRIF contain curated text annotations describing disease–gene associations. These can be queried with text-mining tools and data can be extracted. The different sources were considered individually and jointly. All presented results refer to the full set of disease associated genes irrespective of the original data source. The number of genes linked to each of the diseases can be seen in row: "Disease Genes" in Table 7.

Since not all disease genes are expressed in the synapse, we used the synaptic proteome (Section "Comparison with proteomic data") to filter the disease associated genes for genes that are expressed in the synapse (see row: "Disease genes in the synapse", Table 7). Since almost all modelled genes are expressed in the synapse we only present numbers describing the overlap between disease proteins found in the synapse and modelled genes (see row "Disease Genes in Synapse and in Modelled Genes", Table 7)

There seem to be large differences in the number range of genes associated with diseases. However, the proportions of genes associated with a disease and expressed in the synapse range between 33% (Bipolar Disorder and Major Depressive Disorder) and 45% (Schizophrenia). When looking at the overlap of modelled genes and

Table 7. Overlap of modelled and disease genes.

Disease	AD	ADHD	Autistic Disorder	Bipolar Disorder	MDD	PD	Schizophrenia
Disease Genes	1511	665	575	1140	616	620	1844
Disease Genes in the Synapse	645 (43%)	233 (35%)	255 (44%)	379 (33%)	202 (33%)	262 (42%)	828 (45%)
Disease Genes in Synapse and in modelled Genes	63 (9.8%)	20 (8.6%)	30 (11.8%)	45 (11.9%)	23 (11.4%)	16 (6.1%)	92 (11.1%)

Overlap of modelled and disease genes and their presence in the synapse and our modelled gene set. Disease information is based on GeneRif, OMIM and EnsemblVariation database data. "AD" stands for Alzheimer's Disease, "ADHD" for Attention Deficit Hyperactivity Disorder and "PD" for Parkinson's Disease. Numbers in brackets refer to the percentages. Percentages in the "Disease Genes in the Synapse" column are relative to the total of "Disease Genes" and "Disease Genes in Synapse and in Modelled Genes" is relative to the number of "Disease Genes in Synapse".

disease-associated genes (in the synapse) numbers vary. Schizophrenia seems to have the highest net overlap (92 genes), but also shows the highest number of total associated genes (1844). In total, between 6.1% (Parkinson's Disease) and 11.8% (Autistic Disorder) of disease genes associated with any of the selected diseases expressed in the synapse appeared in at least one model.

Table 8. Modelled genes associated with three or more of the selected diseases.

GeneNames	ADHD	AD	Autistic Disorder	Bipolar Disorder	MDD	Schizophrenia	PD
CACNA1C, *DRD2*, *GRIN2A*, *GRIN2B*	1	1	1	1	1	1	1
GRM5	1	1	1	1	0	1	1
CACNB2, *DRD1*	1	1	1	1	1	1	0
HOMER1	0	1	1	0	1	1	1
CACNA1S, *GRM7*	1	0	1	1	1	1	0
NOS1	1	1	0	1	0	1	1
GNB3, *GRM2*	0	1	0	1	1	1	0
GRIA2	0	1	1	0	1	1	0
GNAL	1	0	0	1	1	1	0
PLA2G6	0	0	0	1	0	1	1
ATP2A3, *CACNA2D1*, *GRM3*	0	0	0	1	1	1	0
GRIK2, *GRM8*, *GRIP1*, *PPP1R1B*	0	0	1	1	0	1	0
DLG4, *NRGN*	0	1	0	0	0	1	1
GRIA4	0	1	0	0	1	1	0
FYN, *GRIA1*, *GRIN1*, *GRM1*, *GNB2L1*	0	1	0	1	0	1	0
SHANK3	1	0	1	0	0	1	0

We were also interested in synaptic genes common to a number of diseases. Table 8 shows the 32 synaptic genes linked to three or more of the diseases included in the analysis. Seven genes are associated to six or all seven tested diseases. The top coverage disease associated genes, found in models annotated, include the protein family voltage-dependent calcium channel family *CACNA1C* and *CACNB2* and dopamine D1 and D2 receptors (*DRD1*, *DRD2*), the inotropic glutamate NMDA receptors, type subunit 2A and 2B (GRIN2A, GRIN2B) as well as the glutamate metabotropic receptor 5 (GRM5). Of the set of modelled genes, 130 (around 50% of the total) are not associated with any of the seven diseases.

In summary, the fraction of genes modelled is relatively small and might indicate that it is challenging to use existing models to make disease predictions. On the other hand the modelled genes can be starting points to extend models to obtain better disease insights, as will be considered later (Approaches to including non-modelled disease genes in models).

Family trees of entities

Our identification of entities in models makes it possible to query in which models a particular entity is contained. The mapping of entities to genes allows querying models by genes that are, or may be, modelled. It is also desirable to query models by families of molecules. For example Gutierrez-Arenas et al. [18] and Nair et al. [17] include *PDE4A*, whereas Kim et al. [30] and Oliveira et al. [28] include *PDE4B* in their models, and Kim et al. [21] and Qi et al. [19] specify *PDE4*. It would be desirable to be able to search for models containing any of the *PDE4* subfamily of genes.

To enable query by class or family, we determined 29 hierarchical family trees of "proteins", "protein families" and "protein multimers" implied by the sets of genes corresponding to each (Fig 10). Each "protein family" or "protein multimer" entity is the parent to one or more "proteins" or "protein families". Each child corresponds to a subset of the proteins in the parent. Tree structures were generated for all "protein multimers" and for "protein families" where a member of that family has been modelled explicitly in at least one of our analysed models. This meant that, for example, PP1 is not represented, since none of its children *PPP1CA*, *PPP1CB* and *PPP1CC* appear in any model explicitly. Individual proteins appear only if they are part of a family or multimer, and they appear in a model – thus, for example, *GRIA4* and *GRIN3* do not appear. Proteins that do not belong to a family, e.g. *PSD95* (*DLG4*), are not shown.

Any entity that is part of a family can be mapped to the root node of its tree. Entities that do not belong to a family are implicitly their own root. This mapping of "entities to entity families" (Fig 3) can be applied to the model-entity catalogue (Fig 4) to give the simplified summary mapping of models to 104 family roots shown in Fig 11. This facilitates comparison of entities across models trying to address the differences in model detail between models.

Frequency of modelling

To give an indication of which are the frequently modelled entities and families of entities, we determined the number of models in which each of the root entities in Fig 12 appears (Table 9). About 50% of root entities appear only in one model. In total, 26 (about 25%) of the entity roots were included in five models or more. The three most frequently modelled entities and families are CaM, CaMKII and Ca, which are included in 18, 22 and 23 out of 30 analysed models respectively. This is due to a number of models focusing specifically on the Ca–CaM–CaMKII pathway or including it as a model part, reflecting its central role in synaptic biology. These top coverage families are followed by families such as PP3 and PP1, PKA and PPP1R, which are also included in the models that include the phosphorylation-dephosphorylation circuit (Section "Models of hippocampal synaptic signalling pathways"). Receptor related families such as AMPAR appear with lower frequency, reflecting the fact that, while crucial for synaptic physiology, not all models include them as a readout mechanism for LTP and LTD. Even though our coverage of models is not complete, it seems likely that cataloguing further models will not change the order much.

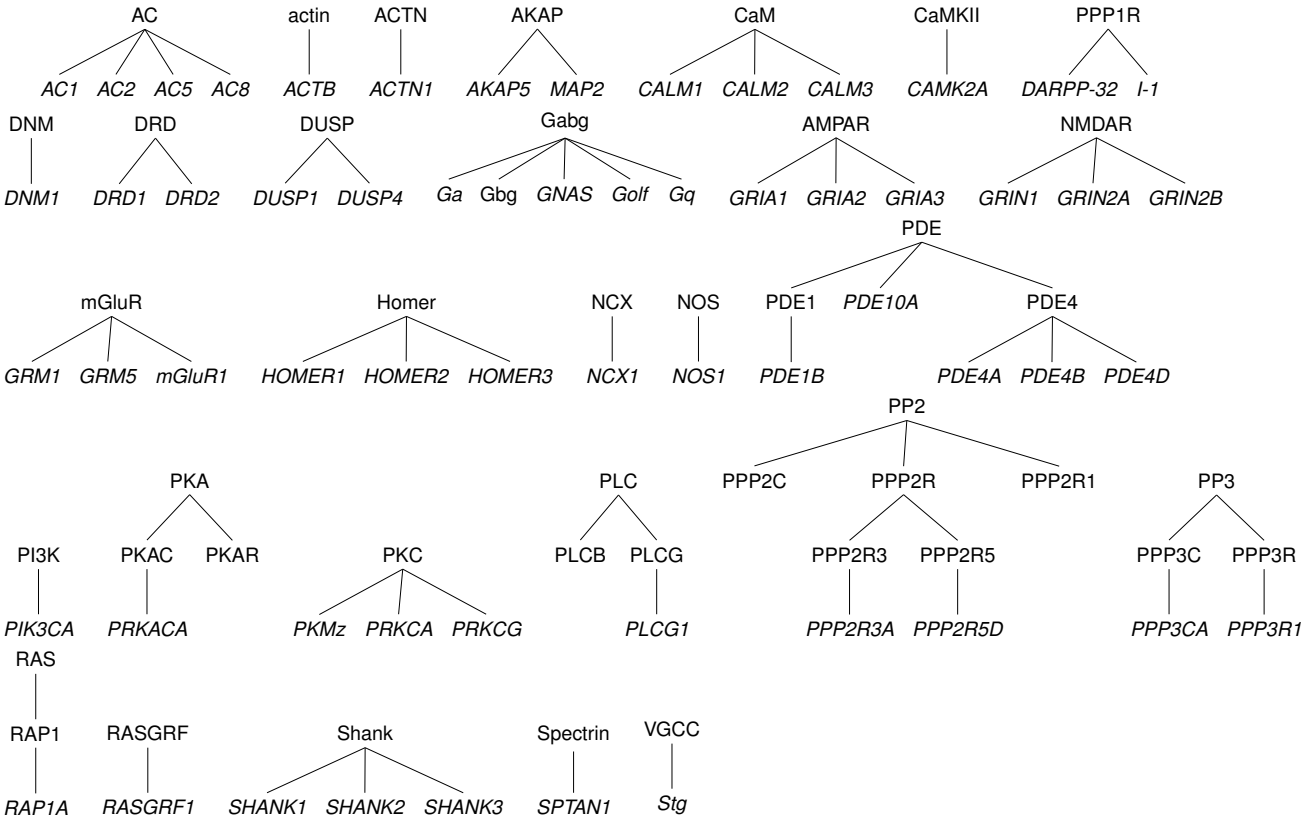


Fig 10. Family trees of "protein families" and "protein multimers". "Proteins" are shown in italics; "protein families" and "protein multimers" in roman. "Proteins" that do not belong to any family are not shown. Only proteins that are specified in models are shown.

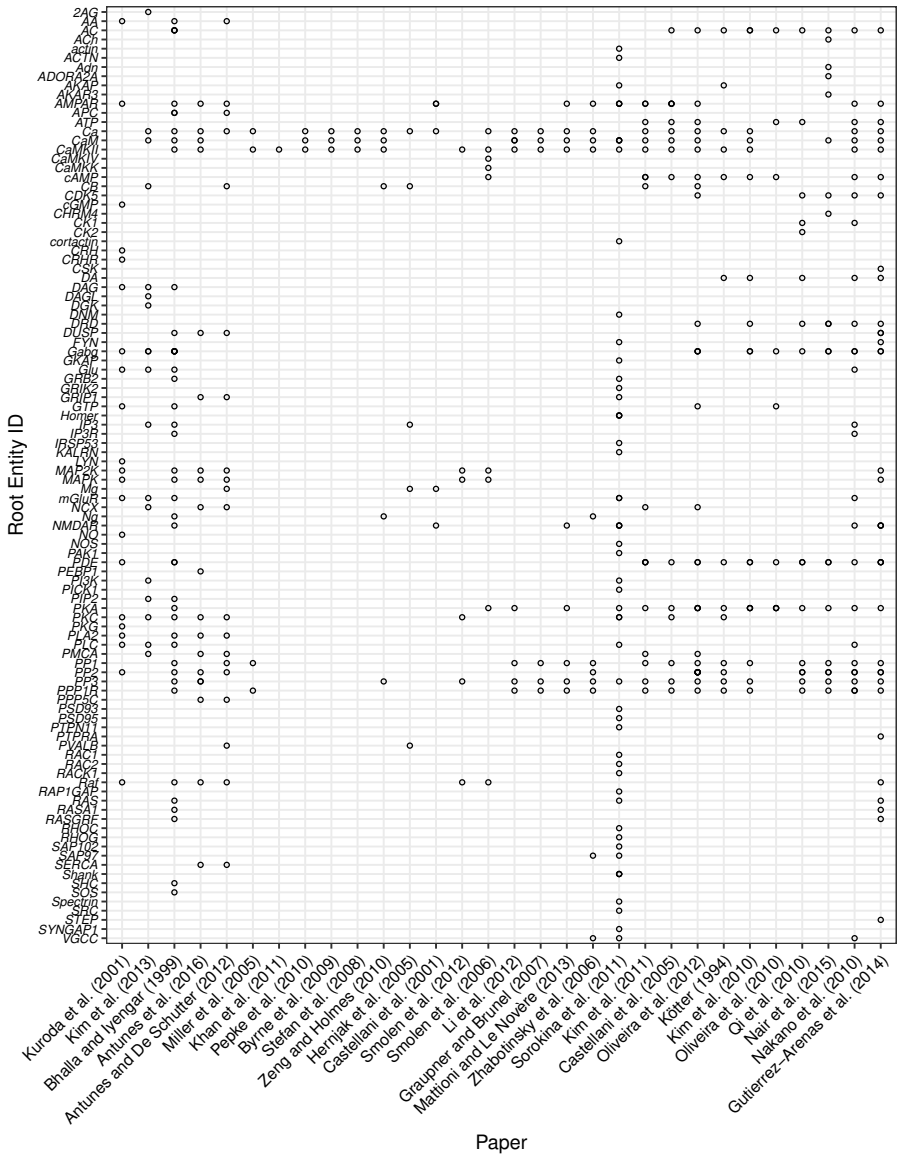


Fig 11. Summary mapping of entities in models. The occurrence of an root entity in a model is indicated by open circles. Lower-level entities are folded into their root entity.

Table 9. Numbers of entities or entity families found in models.

Entity family	Models	Frequency	% Frequency
2AG, actin, ACTN, Adn, AKAP, AKAR3, CaMKIV, CaMKK, cGMP, CHRM4, cortactin, CRH, CRHR, CSK, DAGL, DGK, DNM, GKAP, GRIK2, Homer, IRSP53, KALRN, LYN, NO, NOS, PAK1, PEBP1, PICK1, PSD93, PSD95, PTPN11, PTPRA, RAC1, RAC2, RACK1, RAP1GAP, RHOC, RHOG, SAP102, Shank, SHC, SOS, Spectrin, SRC, STEP, SYNGAP1	1	46	47.4
APC, CK1, FYN, GRB2, IP3R, PI3K, PIP2, PVALB, RASA1, RASGRF, SAP97, SERCA	2	12	12.4
AA, DAG, GRIP1, Mg, Ng, RAS, VGCC	3	7	7.2
CDK5, DUSP, Glu, GTP, IP3, PLA2	4	6	6.2
DA, DRD, mGluR, NCX, PLC, PMCA	5	6	6.2
CB, NMDAR	6	2	2.1
ATP, MAP2K, MAPK, Raf	7	4	4.1
cAMP, Gabg, PKC, PP2	9	4	4.1
AC	10	1	1.0
AMPAR, PDE	12	2	2.1
PPP1R	14	1	1.0
PKA	15	1	1.0
PP1	16	1	1.0
PP3	17	1	1.0
CaM	18	1	1.0
CaMKII	22	1	1.0
Ca	23	1	1.0

"Models" is the number of models containing the entity or at least one member of the family. "Frequency" is the number of appearances of the family or entity in the given number of models, and "% Frequnecy" is the frequency expressed as a percentage.

Comparing models based on their entities

Having annotated the models with entities enabled us to compare models with each other by applying a hierarchical clustering approach to the model-entity root mapping (Fig 11). Ward's 2D method, as implemented in R's `hclust` function was used to give the dendrogram shown in Fig 12. We also applied the clustering to the full model-entity matrix (Fig 4), with similar results, though slightly less meaningful groupings.

In Fig 12 similar models cluster together. Three models (Byrne et al. [58], Pepke et al. [20] and Stefan et al. [59]) are clustered together as they all contain the identical set of entities: Ca, CaM and CaMKII. The closely related model of Zeng and Holmes [27] includes CB as well, and the closely related models of Miller et al. [49] and Khan et al. [32] are also centred on CaMKII. The related models of Smolen et al., 2006 [100] and Smolen et al., 2012 [16] feature the MAPK pathway, in addition to CaMKII.

The group of models containing Li et al. [33], Graupner and Brunel [22], Mattioni and Le Novère [34] and Zhabotinsky et al. [83] are all variations on the CaMKII phosphorylation-dephosphorylation circuit, all adding PP1 and PP3 (calcineurin) to the Ca–CaM-CaMKII pathway. All the models so far are hippocampal; Kim et al. [31] is the closest related striatal model to those mentioned. The model of Sorokina et al. [23] is dissimilar to other models, reflecting the large number of entities, particularly scaffolding proteins, which are contained in this model but not in others.

The next cluster contains a sub-cluster of mostly striatal models [15, 17–19, 21, 30], with the exception of Castellani et al. [82], which is one of the few hippocampal models

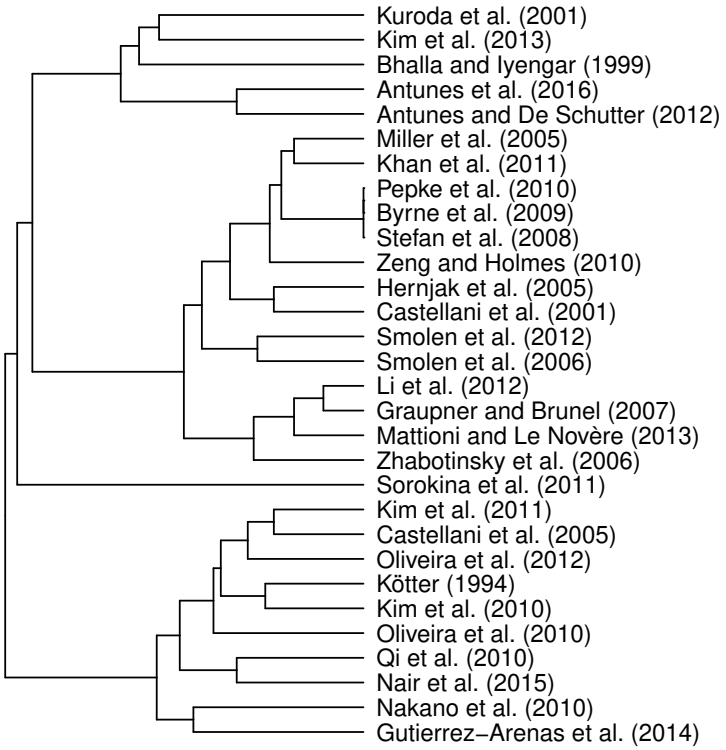


Fig 12. Clustering of model-entity family root matrix. Clustering as implemented in R's `hclust` function with the `Ward.2D` method.

to contain the AC–cAMP–PKA pathway as well as hydrolisation of cAMP to AMP by PDE. The model of Bhalla and Iyengar contains these pathways and many more, accounting for its loose connection with this cluster. In summary, we have shown that models the entity composition can be used to the similarities between models.

Approaches to including non-modelled disease genes in models

Knowing which disease associated genes are included in models helps models with high potential to explain disease impact on the synapse to be identified ("Modelled genes and their overlap with disease genes"). It also allows us to identify disease associated proteins which do not appear in the models we analysed. Of all disease associated genes, 1,248 are found in the synaptic proteome but not in any of the analysed models. Table 10 shows the 32 genes that are associated with 5, 6 or all 7 diseases, and which do not appear in any of the investigated models. Of these, *COMT* and *SLC6A3* are associated with all 7 diseases of interest. Since these genes are associated with all or many studied diseases, they could be of interest when it comes to gaining a better understanding of generic disease dysfunctions.

Supporting the idea that genes implicated in many diseases could be potentially targets for modelling, we identified two genes, *COMT* and *MAOA*, that have been included in metabolic models [137, 138]. Functionally, the catechol O-methyltransferase

(*COMT*) degrades catechols, such as dopamine, by catalysing their methylation. This methylation results in one of the major degradative pathways of the catecholamine transmitters [139]. Dopamine is included in a number of analysed models [140, 141], and it could be possible to explore what happens in these models if there is an excess of dopamine due to *COMT* malfunction.

Table 10. Disease associated genes not appearing in any of the annotated models.

Gene Names	ADHD	AD	Autistic Disorder	Bipolar Disorder	MDD	Schizophrenia	PD
COMT, *SLC6A3*	1	1	1	1	1	1	1
GIGYF2	1	0	1	1	1	1	1
GSK3B, *ABCB1*	1	1	0	1	1	1	1
ANK3, *ENO1*, *KIF5C*, *MAOA*, *PRNP*, *SLC17A6*, *CSMD1*	1	1	1	1	1	0	1
ACE, *GAD1*	0	1	1	1	1	1	0
DDC, *FMR1*	1	0	1	1	0	1	1
APAF1, *DFNA5*, *ELAVL2*, *GRIK1*, *HINT1*, *ITIH1*, *ITIH3*, *ITIH4*, *STT3A*, *LIG4*, *NDUFAB1*, *NDUFB7*, *NPY*, *NTRK3*, *GATB*, *SMARCA2*, *MAD1L1*, *PRPF3*, *SH3PXD2A*, *TRANK1*, *PPIF*, *NT5C2*, *KIF21B*, *RPRD2*, *SYNE1*, *NGEF*, *TENM4*, *GNL3*, *MPP6*, *MRPS21*, *RAB39A*, *CNNM2*, *OXR1*, *ANKS1B*, *VARS2*, *AS3MT*, *PALB2*, *DCTN5*, *PPP1R21*, *MTPN*, *SLC39A12*, *CHSY3*	1	0	1	1	1	0	1
CNR1	1	1	0	0	1	1	1
YWHAZ	1	1	1	0	0	1	1
SNAP25	1	1	1	0	1	0	1
CNTNAP2	1	1	1	1	0	0	1

The table only lists genes that are associated to four or more diseases.

Genes associated will all studied diseases could represent generic disease mechanisms, in which case exploring the role of *COMT* in dopaminergic models would indicate the possible influence of the gene in many diseases. An alternative approach is to consider disease specific genes not appearing in models and associated to only one of the selected diseases. Integrating such proteins into pre-existing models could thus help to gain disease-specific insights. 824 of the disease associated genes are specific to one disease only. To identify genes that can be integrated into existing models, the list of non-modelled disease associated genes was compared with genes in pathways enriched amongst the modelled genes.

For example, all disease genes unique to Schizophrenia were compared with the list of genes in pathways significantly enriched amongst the modelled genes, giving a list of 8 genes, each of which is found in one or more pathways (Table 11). One of these genes is *LAMTOR2*. The LAMTOR2:LAMTOR3 complex binds MAPK components [142], together with other members of the *MAPK2* and *MAPK* activation pathway, such as *RAF1*, *MAPK1*, *MAPK3* and *MAP2K2*. In this role it contributes to the activation of the MAPK pathway which has a central role in striatal and cerebellar synapses. Including the influence of *LAMTOR2* on the activity of MAPK in a pre-existing model could hence help to better understand its role and links to and effects on schizophrenia. Integrating *LAMTOR2* activity in the model could be done mechanistically, or

Table 11. Schizophrenia specific genes not found in models and appearing in pathways that are enriched in annotated models.

Gene Name	Gene Name (long)	REACTOME pathway	Pathway ID
CCK	cholecystokinin	G alpha (q) signalling events	R-HSA-416476
LAMTOR2	late endosomal/lysosomal adaptor, MAPK and MTOR activator 2	MAP2K and MAPK activation, FCERI mediated MAPK activation, VEGFR2 mediated cell proliferation, RAF/MAP kinase cascade	R-HSA-5674135, R-HSA-2871796, R-HSA-5218921, R-HSA-5673001
PSMB1	proteasome subunit beta 1	FCERI mediated MAPK activation, VEGFR2 mediated cell proliferation, RAF/MAP kinase cascade	R-HSA-2871796, R-HSA-5218921, R-HSA-5673001
PSMB4	proteasome subunit beta 4	FCERI mediated MAPK activation, VEGFR2 mediated cell proliferation, RAF/MAP kinase cascade	R-HSA-2871796, R-HSA-5218921, R-HSA-5673001
PSMC1	proteasome 26S subunit and ATPase 1	FCERI mediated MAPK activation, VEGFR2 mediated cell proliferation, RAF/MAP kinase cascade	R-HSA-2871796, R-HSA-5218921, R-HSA-5673001
PSMC4	proteasome 26S subunit and ATPase 4	FCERI mediated MAPK activation, VEGFR2 mediated cell proliferation, RAF/MAP kinase cascade	R-HSA-2871796, R-HSA-5218921, R-HSA-5673001
PSMD2	proteasome 26S subunit and non-ATPase 2 and	FCERI mediated MAPK activation, VEGFR2 mediated cell proliferation, RAF/MAP kinase cascade	R-HSA-2871796, R-HSA-5218921, R-HSA-5673001
TUBB3	tubulin beta 3 class III	Chaperonin-mediated protein folding	R-HSA-390466

functionally, for example by influencing the MAPK concentration.

Discussion

We have developed a catalogue of genes whose corresponding proteins correspond to entities in computational models of synaptic plasticity. To achieve this we developed a new set of standard identifiers for entities in computational models, and mapped those entities corresponding to proteins and protein families onto genes. Although time and lack of machine-readable model descriptions constrained the number of models we could analyse, by selecting models from three brain regions (hippocampus, striatum and cerebellum) we are confident that we have covered the bulk of proteins in models.

We were able to identify 294 genes that could be mapped to entities in computational models. This corresponds to 4.2% of the 6,706 known genes in the synaptic proteome. Enrichment analysis showed that, compared to the set of proteins found in the synapse, the genes in models tended to have more signalling functions, which reflects the focus on signalling pathways in such models. This suggests considerable scope for including new molecules in models. However, models of synapses at the molecular level are already complex and are beset by problems of determining parameters. One strategy to prioritise molecules to add to models is to add those most relevant for disease. Our comparison of the list of genes in models with databases of gene-disease association shows that many disease-associated genes are not currently included in synaptic models, and suggests targets for future modelling.

Targeting disease-relevant proteins for modelling

The genes in models are more associated with neurological diseases, such as Schizophrenia, Alzheimer's, Huntington's disease and bipolar disorder, than randomly selected genes in the synaptic proteome or the whole genome. Nevertheless, depending on the disease, the number of disease-associated genes included in models range between 6% and 12% of the disease-associated genes in the synapse. This suggests that there is considerable potential to include disease-related genes in models. Including these molecules could make these models more useful in helping elucidate disease mechanisms and helping to identify new drug targets.

We identified two un-modelled genes associated with 7 neurological diseases, *COMT* and *MAOA* and we found they have close functional links with existing models. By incorporating pathway enrichment results, we identified *LAMTOR*, a gene uniquely associated with Schizophrenia. *LAMTOR* is linked to the MAP kinase pathway, which features in a number of existing models. This demonstrates the utility of our approach for identifying which proteins to incorporate in existing models so that they can make disease-associated predictions. Further investigation using this approach could indicate other target proteins to add to existing synaptic pathway models to make them more informative about the influence of diseases on the synapse.

A new ontology for computational neuroscience models

The challenge we faced mapping model entities to genes highlighted a gap between bioinformatics, where each gene is well-defined and has a commonly used identifier, and computational neuroscience, where the elements of models are defined at varying levels of precision: for example they may be proteins, protein families or multimers of proteins. Even within the same model, one element may be specified precisely, for example a particular isoform (PKMζ), and another element may be generic, for example "plasticity related proteins" [16]. From a bioinformatics perspective this may seem offensive, but from the viewpoint of computational neuroscience it is entirely valid: a computational model can be seen as a means to reasoning about a hypothesis; the formulation of the model is the hypothesis and the simulations embody the reasoning that generates the predictions arising from the hypothesis [143]. The modelling process sometimes even requires hypothetical elements, which have no existing identifier. For example, one seminal computational neuroscience model [144] contained hypothetical elements ("gating particles") that predicted essential features of ion channels function.

The problem of mapping model constituents onto biological entities was noted by the originators of the MIRIAM standard [118]. This standard suggests solving the problem of mapping entities at different levels of abstraction by using a "HasVersion" qualifier to map reactants in models to multiple entities, e.g. to map IP3R to Inositol 1,4,5-triphosphate receoptors type 1, 2 and 3. Most of the models we investigated had not been annotated to MIRIAM standards, and we found it more efficient to define our own ontology containing proteins and protein families. We found that existing ontologies such as UniProt, HGNC gene families [145] and Neurolex [146] were not extensive enough to map proteins specified at different levels of precision (e.g. PDE4A, PDE4) to common families (e.g. PDE), though HGNC gene families covered about half of the protein families we identified.

In the absence of a suitable ontology, we used HGNC gene families and curated other family relationships manually to give a full list of entities (Supporting Information Tables S1) and mappings of proteins to families and multimers in which they occur (Supporting Information Tables S3, S4). These tables form the kernel of an ontology, and we have demonstrated that it can be used to determine the potential genes underlying the proteins in computational models, and to cross-link these genes with

expression data. Furthermore, we have demonstrated that the ontology can be used to compare models, for example using hierarchical clustering, and to summarise of how often various protein families have been modelled. By annotating models with identifiers of brain region or neuron type, the set of possible proteins belonging to a model could be narrowed down according to the genes that are expressed in a given region. The same procedure could be used to link the genetic content of synaptic models with other types of data, for example spatial expression data from the Allen Brain atlas. This would make it possible to check that a particular model was valid in the brain region it is supposed to represent, or, conversely, could be used to find brain regions for which a particular model might be valid.

The number of models analysed in this paper was limited by the time it took us to annotate models we had not constructed. While some repositories, such as the curated branch of BioModels, enforce curation of models to MIRIAM standards [118], it would be desirable for all models to be annotated consistently at the time of publication or deposition in a repository. Annotation would be a fairly quick process for authors familiar with the models, and the quality of the information would be higher than if annotated by third parties. Three of the 30 models we investigated were annotated to MIRIAM standards. We did not use the MIRIAM annotations of these models, partly so that our annotation of models was consistent and partly because the MIRIAM standard suggests mapping to external identifiers that are often at a finer level of granularity than we needed to compare models to proteomic data. Were more models curated to MIRIAM standards, it would be worthwhile developing a mapping to our identifiers.

As discussed above, some models are of necessity not precise about which protein is specified. To address this, one option would be for the computational neuroscience and bioinformatics communities to adopt an ontology along the lines of the ones we have generated here. If the ontology were stored in the Interlex dynamic lexicon of biomedical terms, a development of Neurolex [146], it would be straightforward for authors to suggest new terms or relationships. The model metadata could be stored by adding fields to existing repository schema, or our data could be converted to a standalone, API-enabled database.

Nomenclature

The nomenclature we have used for entities has been decided by the authors. We have been guided by gene names, and some of our choices might be controversial, for example naming PP2B (calcineurin) PP3. Our rationale for using identifiers related to gene names is so there is more consistency between the names of members in a family. For example, in Fig 10, PP3 is the parent of the catalytic and regulatory subunits PPP3C and PPP3R; having PP2B as a parent would not be equally consistent. It would be desirable for the computational neuroscience and bioinformatics communities to agree a common nomenclature.

New directions in modelling

We have demonstrated the potential of our method of identifying entities in models and mapping them to genes to suggest new, disease-relevant directions for modelling. We believe there is considerable potential for the work to be adopted to suit the needs of the community. Our files are available (S1 File) and suggestions for additions or amendments are welcome. [We will also be making our files available via github.]

More speculatively, despite the challenge of expanding the number and relevant proteins in models of synaptic plasticity, we believe that the time has come to incrementally increase the number of proteins involved in models, especially those involved in disease mechanisms.

Methods

Identifying entities in models

The question of what entities mean is outlined in "Analysis of proteins in synaptic models", subection "Identifying entities in models". The constituent entities of each model were identified by one of the authors (EMW, KFH or DCS) reading the paper, or extracting elements from a machine-readable representation of the model, for example CellML or Kappa descriptions in the cases of Bhalla and Iyengar [80] and Sorokina et al. [23] respectively. The name used to identify the entity in the model was then mapped to the standardised list of entities that we built up as we looked through the models. In some cases model entities were not specified enough to allow us to map them unambiguously onto a model entity – for example "Plasticity Related Protein" [16]. We did not consider a complex as an entity – for example a Ca-CaM-CaMKII complex would give rise to Ca (ion), CaM ("protein") and CaMKII ("protein multimer"). In naming our standard entities, we have tried to use names commonly used in models, but for entities that have not appeared in many models we have tended to use the newer standard names that appear in the NCBI or UniProt databases.

Mapping entities to a unique gene identifier

To obtain a common identifier for all entities we searched for an ontology that could be used to identify our entities, especially "protein families" and "protein multimers". We considered a number of potential ontologies:

The Computational Neuroscience Ontology (`http://bioportal.bioontology.org/ontologies/CNO`) This ontology covers the description of the modelling technique (e.g. Integrate-and-fire neurons) rather than the components of the model.

HGNC Gene families (`http://www.genenames.org/`) The Human Gene Organisation Gene Nomenclature Committee (HGNC) approves unique symbols and names for human genes, and also places genes in families, based on characteristics such as function, homology, domains and phenotype [145]. Placing genes into families is a manual process, often involving specialists who are expert in that family of genes. Often, but not always, genes in the same family have a common root symbol. The process of defining families is ongoing.

InterPro protein families (`http://www.ebi.ac.uk/interpro`) The InterPro Consortium is a federation amalgamating protein signature databases (Gene3D, Conserved Domain Database, HAMAP, PANTHER, Pfam, PIRSF, PRINTS, ProDom, PROSITE, SMART, SUPERFAMILY, Structure-Function Linkage Database and TIGRFAMs) [147]. Protein signatures are predictive models build on fragments of amino acid sequences that share local features (e.g. conservation at different positions) known to be associated with a function or structure [148]. There are multiple computational approaches that are detecting such patterns and define types of signatures [149]. The similarity in signature matches between proteins is used to define a hierarchy of families.

Manual NCBI search (`www.ncbi.nlm.nih.gov/gene/`) The National Center for Biotechnology Information (NCBI) provides access to biomedical and genomic information. We used their searchable database of genes, which can be queried with a number of different identifiers.

We intended to map out entities using information supplied by one of these ontologies, but no one source proved sufficient. In InterPro, there are a number of families that correspond exactly to proteins, for example Phospholipase A2 (IPR001211) and Phosphoinositide phospholipase C (IPR001192). However, some proteins, including SOS1 and SOS2, belong to very broad families.

In the HGNC database we identified a relatively large number of our entities that correspond to existing HGNC gene families. For example the HGNC Homer family (short for "Homer scaffolding proteins") comprises the genes *HOMER1*, *HOMER2* and *HOMER3* and the genes *PPP3CA*, *PPP3CB*, *PPP3CC*, *PPP3R1* and *PPP3R2* belong to the HGNC PP3 family (short for "Calcineurin"). Other entities do not correspond to a single gene family, but can be extracted from the database by selecting multiple families. For example SHANK, by which we mean the family of proteins encoded by *SHANK1*, *SHANK2* and *SHANK3* may be selected from the gene families list by selecting all genes that are in the "Ankyrin repeat domain containing" (ANKRD) and "PDZ domain containing" (PDZ) gene families. Some of our entities cannot be recovered by searching for families. For example SOS (by which we mean the proteins encoded by *SOS1* and *SOS2*) are in both the "Rho guanine nucleotide exchange factors" and "Pleckstrin homology domain containing" families, but so are 35 other proteins.

We also curated our own mappings by manually querying the NCBI portal by searching for human genes matching a full protein name and a common gene prefix, suffix or infix, if available. For example, Entrez IDs for a "protein family" of Voltage-dependent calcium channel were obtained with the following query: 'Voltage-dependent calcium channel[All Fields] AND CACN*[All Fields] AND "Homo sapiens"[Organism]'. The top 20 results were considered and only entries with the closest description and gene summary to the search term were extracted.

Although we were not able to map all our entities by relying on only one ontology, we found that HGNC families covered more of our entities than Interpro, so we used this as a basis for developing an ontology to describe the molecular components of computational neuroscience models. We tried to map all entities of type "protein family" and "protein multimer" to HGNC families. Manual NCBI mappings were used to check and verify that HGNC families represented the modelled group of genes.

In situations where we were unable to find a corresponding HGNC family we (1) suggested some protein groups to be added to the list of HGNC families and await approval of the request; (2) we had no choice but to fall back on our manual NCBI mapping. The combination of the above lead us to our final mappings. S3 Table and S4 Table show identified HGNC families as well as the genes belonging to them. The superscript given with the HGNC family name indicates its origin, the official HGNC mapping vs. custom mapping. The columns "IN.SYNAPSE" and "OUT.SYNAPSE" are explained in Section "Comparison with proteomic data".

Enrichment Analysis

A commonly used method to find statistically significant commonalities between large gene lists is enrichment analysis, also known as over-representation analysis. Based on information contained in ontological databases, enrichment analysis can show if a set of "genes of interest" contains a significantly high number of genes with the same annotation. This approach allows us to gain a better understanding of underlying common themes in our "genes in models" list.

The underlying principle of such an enrichment analysis is to estimate, for each specific category annotated in the database of interest, if the number of genes in our genes of interest set associated with a certain category is larger than expected by chance. To test this relationship statistically, the hypergeometric distribution or one-tailed Fisher's exact test is commonly applied. Both are known to be equivalent [150].

The four key numbers required to carry out the statistical calculations are:

1. The number of elements in the full dataset, also considered as the background dataset, N. In our case these are all proteins part of the synaptic proteome.
2. The number of elements n in the subset of the full dataset which is tested for enrichment. This is the number of genes in the "genes in models" list.
3. The number of elements associated to a certain trait in the full dataset, T. It corresponds to the set of genes annotated to any term in one of the databases, e.g. "Schizophrenia", which describes a disease in the DO database.
4. The subset of n shared by the elements found in T, denoted as t. This refers to the number of genes within a category that are also present in our "genes in models" list.

The probability of encountering the exact number of hits t of interest given N, n and T is calculated with the hypergeometric probability $h(t; N, n, T)$:

$$h(t; N, n, T) = \frac{\binom{T}{t}\binom{N-T}{n-t}}{\binom{N}{n}} \tag{1}$$

To describe the probability of finding greater than or equal to the number of items of interest t, we use the cumulative hypergeometric probability:

$$p(t; N, n, T) = \sum_{x=t}^{T} h(x; N, n, T) = \sum_{x=t}^{T} \frac{\binom{T}{x}\binom{N-T}{n-x}}{\binom{N}{n}} \tag{2}$$

If this probability is less than a criterion (e.g. $p < 0.01$), the dataset is regarded as enriched [150] for the tested category.

For the analysis, ontology terms for all genes in the background dataset N were obtained. Initially two background sets were considered, containing (1) all genes in the genome and (2) all proteins found in the synapse. Since results were quite similar and the focus of this study is on the synaptic region rather than the whole organism, we only present results obtained with the second dataset as the background set of genes.

We analysed all terms that had at least one gene associated to our "genes in models". For each such term, the p-value was calculated, indicating potential enrichment, and then corrected for multiple comparison, using the Benjamini and Yekutieli [151] method. Terms with adjusted p-values smaller than 0.01 are presented in the final results.

`topONTO` and `topGO`

Ontologies that supply functional annotation information are organised in a hierarchical structure, with the most generic terms at the top, and the most specific ones at the bottom. The higher the term is located in the hierarchy, the more genes are associated to it as it aggregates all genes from its child terms. Hence, a single gene can be found on different levels of annotation specificity. Depending on the purpose of the analysis it is important to be able to choose the level of retrieved terms.

To retrieve the most specific and refined terms among significantly enriched ones, we used an algorithm proposed by Alexa et al. [152] and implemented for the GO database by the R *topGO* package. Since GO is represented as a Directed Acyclic Graph (DAG), the authors incorporated the underlying GO graph topology in the term scoring approach, removing strong correlations commonly occurring between high level terms. This allows the enrichment of a very generic term to be ignored, and less frequent but more specific and potentially more interesting low level ones to be identified.

Assuming that a child term is potentially more interesting than its more generic ancestors, significance of a term is calculated depending on its child terms. Out of multiple versions implementing this idea, we used the *elim* algorithm paired with Fisher's exact test. The decision was based on the clear number of comparisons conducted by the algorithm. This number was further used to correct for the false discovery rate.

In the *elim* approach [152], enrichment analysis starts at the bottom of the ontology graph. If a child term is significantly enriched amongst the genes of interest, this influences the number of genes annotated to its ancestor terms. All genes associated to the enriched child term are removed from the ancestor terms leaving most specific ones with the minimal indicated significance.

We discovered that the algorithm leads to more refined results than a set-based enrichment analysis that ignores the ontology structure. Therefore, we were interested in applying a same approach to other gene annotation sets. This can be achieved with the *topOnto* R package [135]. It extends the advantage of the Alexa et al. method to any hierarchically structured dataset. Since both REACTOME and DO satisfy this requirement, we were able to apply the same approach to all chosen annotation sets.

Supporting information

S1 File. Data and code. A zip file containing the data tables, and mapping and analysis code that will reproduce the results in this paper.

S1 Table. Full list of entities. List of entities containing the ID, name, type and for proteins, mapping to gene.

S2 Table. Synaptic Proteome Studies. List of synaptic proteome publications and respective datasets used in this study.

S3 Table. Protein family members. List of entities in distinct protein families - "in" and "out" of the synapse.

S4 Table. Protein multimer members. List of entities in distinct protein multimers - "in" and "out" of the synapse.

References

1. Martin SJ, Grimwood PD, Morris RGM. Synaptic Plasticity and Memory: An Evaluation of the Hypothesis. Annu Rev Neurosci. 2000;23(1):649–711. doi:10.1146/annurev.neuro.23.1.649.

2. Bliss TV, Lømo T. Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. J Physiol (Lond). 1973;232(2):331–356.

3. Lynch GS, Dunwiddie T, Gribkoff V. Heterosynaptic depression: a postsynaptic correlate of long-term depression. Nature. 1977;266:737–739.

4. Abbott LF, Nelson SB. Synaptic plasticity: taming the beast. Nat Neurosci. 2000;3:1178–1183.

5. Nadim F, Bucher D. Neuromodulation of neurons and synapses. Curr Opin Neurobiol. 2014;29:48–56. doi:10.1016/j.conb.2014.05.003.

6. Carlisle HJ, Fink AE, Grant SG, O'Dell TJ. Opposing effects of PSD-93 and PSD-95 on long-term potentiation and spike timing-dependent plasticity. J Physiol (Lond). 2008;586(Pt 24):5885–5900. doi:10.1113/jphysiol.2008.163469.

7. Pocklington AJ, Cumiskey M, Armstrong JD, Grant SGN. The proteomes of neurotransmitter receptor complexes form modular networks with distributed functionality underlying plasticity and behaviour. Mol Syst Biol. 2006;2(1). doi:10.1038/msb4100041.

8. Morrison A, Diesmann M, Gerstner W. Phenomenological models of synaptic plasticity based on spike timing. Biol Cybern. 2008;98(6):459–478. doi:10.1007/s00422-008-0233-1.

9. Manninen T, Hituri K, Kotaleski JHH, Blackwell KT, Linne MLL. Postsynaptic signal transduction models for long-term potentiation and depression. Front Comput Neurosci. 2010;4.

10. Nair AG, Gutierrez-Arenas O, Eriksson O, Jauhiainen A, Blackwell KT, Kotaleski JH. Modeling intracellular signaling underlying striatal function in health and disease. Prog Mol Biol Transl Sci. 2014;123:277–304.

11. Blackwell KT, Jedrzejewska-Szmek J. Molecular mechanisms underlying neuronal synaptic plasticity: systems biology meets computational neuroscience in the wilds of synaptic plasticity. Wiley interdisciplinary reviews Systems biology and medicine. 2013;5(6):717–731.

12. Lassek M, Weingarten J, Volknandt W. The synaptic proteome. Cell Tissue Res. 2015;359(1):255–65. doi:10.1007/s00441-014-1943-4.

13. Bayés À, Collins MO, Croning MD, van de Lagemaat LN, Choudhary JS, Grant SG. Comparative study of human and mouse postsynaptic proteomes finds high compositional conservation and abundance differences for key synaptic proteins. PLoS ONE. 2012;7(10):e46683.

14. Faas GC, Raghavachari S, Lisman JE, Mody I. Calmodulin as a direct detector of Ca^{2+} signals. Nat Neurosci. 2011;14(3):301–304. doi:10.1038/nn.2746.

15. Nakano T, Doi T, Yoshimoto J, Doya K. A kinetic model of dopamine- and calcium-dependent striatal synaptic plasticity. PLoS Comput Biol. 2010;6(2):e1000670.

16. Smolen P, Baxter DA, Byrne JH. Molecular constraints on synaptic tagging and maintenance of long-term potentiation: a predictive model. PLoS Comput Biol. 2012;8(8).

17. Nair AG, Gutierrez-Arenas O, Eriksson O, Vincent P, Hellgren Kotaleski J. Sensing positive versus negative reward signals through adenylyl cyclase-coupled GPCRs in direct and indirect pathway striatal medium spiny neurons. J Neurosci. 2015;35(41):14017–14030.

18. Gutierrez-Arenas O, Eriksson O, Kotaleski JH. Segregation and crosstalk of D1 receptor-mediated activation of ERK in striatal medium spiny neurons upon acute administration of psychostimulants. PLoS Comput Biol. 2014;10(1):e1003445. doi:10.1371/journal.pcbi.1003445.

19. Qi Z, Miller GW, Voit EO. The internal state of medium spiny neurons varies in response to different input signals. BMC Syst Biol. 2010;4(1):1–16. doi:10.1186/1752-0509-4-26.

20. Pepke S, Kinzer-Ursem T, Mihalas S, Kennedy MB. A dynamic model of interactions of Ca^{2+}, calmodulin, and catalytic subunits of Ca^{2+}/calmodulin-dependent protein kinase II. PLoS Comput Biol. 2010;6(2):e1000675. doi:10.1371/journal.pcbi.1000675.

21. Kim M, Huang T, Abel T, Blackwell KT. Temporal sensitivity of protein kinase A activation in late-phase long term potentiation. PLoS Comput Biol. 2010;6(2):1–14. doi:10.1371/journal.pcbi.1000691.

22. Graupner M, Brunel N. STDP in a bistable synapse model based on CaMKII and associated signaling pathways. PLoS Comput Biol. 2007;3(11):e221.

23. Sorokina O, Sorokin A, Armstrong JD. Towards a quantitative model of the post-synaptic proteome. Mol Biosyst. 2011;7:2813–2823. doi:10.1039/C1MB05152K.

24. Stefan MI, Marshall DP, Le Novère N. Structural analysis and stochastic modelling suggest a mechanism for calmodulin trapping by CaMKII. PLoS ONE. 2012;7(1):e29406. doi:10.1371/journal.pone.0029406.

25. Hepburn I, Chen W, Wils S, De Schutter E. STEPS: efficient simulation of stochastic reaction–diffusion models in realistic morphologies. BMC Syst Biol. 2012;6(1):36.

26. Hernjak N, Slepchenko BM, Fernald K, Fink CC, Fortin D, Moraru II, et al. Modeling and analysis of calcium signaling events leading to long-term depression in cerebellar Purkinje cells. Biophys J. 2005;89(6):3790–3806.

27. Zeng S, Holmes WR. The effect of noise on CaMKII activation in a dendritic spine during LTP induction. J Neurophysiol. 2010;103(4):1798–1808. doi:10.1152/jn.91235.2008.

28. Oliveira RF, Terrin A, Di Benedetto G, Cannon RC, Koh W, Kim M, et al. The Role Of Type 4 Phosphodiesterases in generating microdomains of cAMP: large scale stochastic simulations. PLoS ONE. 2010;5(7):e11725. doi:10.1371/journal.pone.0011725.

29. Oliveira RF, Kim M, Blackwell KT. Subcellular location of PKA controls striatal plasticity: stochastic simulations in spiny dendrites. PLoS Comput Biol. 2012;8(2):e1002383. doi:10.1371/journal.pcbi.1002383.

30. Kim M, Park AJ, Havekes R, Chay A, Guercio LA, Oliveira RF, et al. Colocalization of protein kinase A with adenylyl cyclase enhances protein kinase A activity during induction of long-lasting long-term-potentiation. PLoS Comput Biol. 2011;7(6):e1002084.

31. Kim B, Hawes SL, Gillani F, Wallace LJ, Blackwell KT. Signaling pathways involved in striatal synaptic plasticity are sensitive to temporal pattern and exhibit spatial specificity. PLoS Comput Biol. 2013;9(3):e1002953. doi:10.1371/journal.pcbi.1002953.

32. Khan S, Zou Y, Amjad A, Gardezi A, Smith CL, Winters C, et al. Sequestration of CaMKII in dendritic spines in silico. J Comput Neurosci. 2011;31(3):581–594.

33. Li L, Stefan MI, Le Novère N. Calcium input frequency, duration and amplitude differentially modulate the relative activation of calcineurin and CaMKII. PLoS ONE. 2012;7(9):e43810+. doi:10.1371/journal.pone.0043810.

34. Mattioni M, Le Novère N. Integration of biochemical and electrical signaling – multiscale model of the medium spiny neuron of the striatum. PLoS ONE. 2013;8(7):e66811. doi:10.1371/journal.pone.0066811.

35. Lisman JE. A mechanism for memory storage insensitive to molecular turnover: a bistable autophosphorylating kinase. Proc Natl Acad Sci USA. 1985;82(9):3055–3057.

36. Kuret J, Schulman H. Mechanism of autophosphorylation of the multifunctional Ca2+/calmodulin-dependent protein kinase. J Biol Chem. 1985;260(10):6427–6433.

37. Miller SG, Kennedy MB. Distinct forebrain and cerebellar isozymes of type II Ca2+/calmodulin-dependent protein kinase associate differently with the postsynaptic density fraction. J Biol Chem. 1985;260(15):9039–9046.

38. Lisman JE, Goldring MA. Feasibility of long-term storage of graded information by the Ca2+/calmodulin-dependent protein kinase molecules of the postsynaptic density. Proc Natl Acad Sci USA. 1988;85(14):5320–5324.

39. Zhabotinsky AM. Bistability in the Ca2+/Calmodulin-dependent protein kinase-phosphatase system. Biophys J. 2000;79(5):2211–2221. doi:http://dx.doi.org/10.1016/S0006-3495(00)76469-1.

40. Petersen JD, Chen X, Vinade L, Dosemeci A, Lisman JE, Reese TS. Distribution of postsynaptic density (PSD)-95 and Ca^{2+}/calmodulin-dependent protein kinase II at the PSD. J Neurosci. 2003;23(35):11270–8.

41. Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J Comput Phys. 1976;22(4):403–434. doi:http://dx.doi.org/10.1016/0021-9991(76)90041-3.

42. Antunes G, Roque AC, Simoes-de Souza FM. Stochastic induction of long-term potentiation and long-term depression. Sci Rep. 2016;6:30899. doi:10.1038/srep30899.

43. Gaertner TR, Kolodziej SJ, Wang D, Kobayashi R, Koomen JM, Stoops JK, et al. Comparative analyses of the three-dimensional structures and enzymatic properties of alpha, beta, gamma and delta isoforms of Ca^{2+}-calmodulin-dependent protein kinase II. J Biol Chem. 2004;279(13):12484–12494. doi:10.1074/jbc.M313597200.

44. Chao LH, Stratton MM, Lee IH, Rosenberg OS, Levitz J, Mandell DJ, et al. A mechanism for tunable autoinhibition in the structure of a human Ca2+/calmodulin- dependent kinase II holoenzyme. Cell. 2011;146(5):732–45. doi:10.1016/j.cell.2011.07.038.

45. Lisman J, Yasuda R, Raghavachari S. Mechanisms of CaMKII action in long-term potentiation. Nat Rev Neurosci. 2012;13(3):169–182. doi:10.1038/nrn3192.

46. Weisstein EW. Necklace; 2017. From MathWorld–A Wolfram Web Resource. Available from: `http://mathworld.wolfram.com/Necklace.html`.

47. Coomber CJ. Site-selective autophosphorylation of Ca^2+/calmodulin-dependent protein kinase II as a synaptic encoding mechanism. Neural Comput. 1998;10:1653–1678.

48. Kubota Y, Bower JM. Transient versus asymptotic dynamics of CaM Kinase II: possible roles of phosphatase. J Comput Neurosci. 2001;11(3):263–279. doi:10.1023/A:1013727331979.

49. Miller P, Zhabotinsky AM, Lisman JE, Wang XJJ. The stability of a stochastic CaMKII switch: dependence on the number of enzyme molecules and protein turnover. PLoS Biol. 2005;3(4):e107+. doi:10.1371/journal.pbio.0030107.

50. Stefan MI, Bartol TM, Sejnowski TJ, Kennedy MB. Multi-state modeling of biomolecules. PLoS Comput Biol. 2014;10(9):e1003844. doi:10.1371/journal.pcbi.1003844.

51. Michelson S, Schulman H. CaM kinase: a model for its activation and dynamics. J Theor Biol. 1994;171(3):281–290. doi:10.1006/jtbi.1994.1231.

52. Gillespie D. Approximate accelerated stochastic simulation of chemically reacting systems. J Chem Phys. 2001;115:1716–1733.

53. Holmes WR. Models of calmodulin trapping and CaM kinase II activation in a dendritic spine. J Comput Neurosci. 2000;8(1):65–85.

54. Le Novère N, Shimizu TS. STOCHSIM: modelling of stochastic biomolecular processes. Bionformatics. 2001;17(6):575–6.

55. Danos V, Feret J, Fontana W, Krivine J. Scalable Simulation of Cellular Signaling Networks. In: Shao Z, editor. Programming Languages and Systems. vol. 4807 of Lecture Notes in Computer Science. Berlin, Heidelberg: Springer; 2007. p. 139–157. Available from: http://dx.doi.org/10.1007/978-3-540-76637-7_10.

56. Faeder JR, Blinov ML, Hlavacek WS. Rule-Based Modeling of Biochemical Systems with BioNetGen. In: Maly IV, editor. Systems Biology. vol. 500 of Methods in Molecular Biology. Humana Press; 2009. p. 113–167. Available from: http://dx.doi.org/10.1007/978-1-59745-525-1_5.

57. Sneddon MW, Faeder JR, Emonet T. Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. Nat Methods. 2011;8(2):177–183. doi:10.1038/nmeth.1546.

58. Byrne MJ, Putkey JA, Waxham NN, Kubota Y. Dissecting cooperative calmodulin binding to CaM kinase II: a detailed stochastic model. J Comput Neurosci. 2009;27(3):621–638. doi:10.1007/s10827-009-0173-3.

59. Stefan MI, Edelstein SJ, Le Novère N. An allosteric model of calmodulin explains differential activation of PP2B and CaMKII. Proc Natl Acad Sci USA. 2008;105(31):10768–10773. doi:10.1073/pnas.0804672105.

60. Crouch TH, Klee CB. Positive cooperative binding of calcium to bovine brain calmodulin. Biochemistry. 1980;19(16):3692–8.

61. Colquhoun D, Lape R. Perspectives on: conformational coupling in ion channels: allosteric coupling in ligand-gated ion channels. J Gen Physiol. 2012;140(6):599–612. doi:10.1085/jgp.201210844.

62. Gaertner TR, Putkey JA, Waxham MN. RC3/Neurogranin and Ca^{2+}/calmodulin-dependent protein kinase II produce opposing effects on the affinity of calmodulin for calcium. J Biol Chem. 2004;279(38):39374–82. doi:10.1074/jbc.M405352200.

63. Hanson PI, Meyer T, Stryer L, Schulman H. Dual role of calmodulin in autophosphorylation of multifunctional cam kinase may underlie decoding of calcium signals. Neuron. 1994;12(5):943–956. doi:10.1016/0896-6273(94)90306-9.

64. Dupont G, Houart G, De Koninck P. Sensitivity of CaM kinase II to the frequency of Ca2+ oscillations: a simple model. Cell Calcium. 2003;34(6):485–497.

65. Gamble E, Koch C. The dynamics of free calcium in dendritic spines in response to repetitive synaptic input. Science. 1987;236(4806):1311–5.

66. Holmes WR, Levy WB. Insights into associative long-term potentiation from computational models of NMDA receptor-mediated calcium influx and intracellular calcium concentration changes. J Neurophysiol. 1990;63(5):1148–1168.

67. Zador A, Koch C, Brown TH. Biophysical Model of a Hebbian synapse. Proc Natl Acad Sci USA. 1990;87:6718–6722.

68. McDougal RA, Hines ML, Lytton WW. Reaction-diffusion in the NEURON simulator. Front Neuroinform. 2013;7(28). doi:10.3389/fninf.2013.00028.

69. Hepburn I, Chen W, Wils S, De Schutter E. STEPS: efficient simulation of stochastic reaction–diffusion models in realistic morphologies. BMC Systems Biology. 2012;6(1):36. doi:10.1186/1752-0509-6-36.

70. Gillespie DT. Stochastic simulation of chemical kinetics. Annu Rev Phys Chem. 2007;58:35–55. doi:10.1146/annurev.physchem.58.032806.104637.

71. Sorokina O, Sorokin A, Armstrong JD, Danos V. A simulator for spatially extended kappa models. Bionformatics. 2013; p. 3105–3106.

72. Kerr RA, Bartol TM, Kaminsky B, Dittrich M, Chang JC, Baden SB, et al. Fast Monte Carlo simulation methods for biological reaction-diffusion systems in solution and on surfaces. SIAM J Sci Comput. 2008;30(6):3126. doi:10.1137/070692017.

73. Andrews SS. Smoldyn: particle-based simulation with rule-based modeling, improved molecular interaction and a library interface. Bionformatics. 2017;33(5):710–717. doi:10.1093/bioinformatics/btw700.

74. Franks KM, Bartol TM Jr, Sejnowski TJ. A Monte Carlo model reveals independent signaling at central glutamatergic synapses. Biophys J. 2002;83(5):2333–48. doi:10.1016/S0006-3495(02)75248-X.

75. Franks KM, Bartol TM, Sejnowski TJ. An {MCell} model of calcium dynamics and frequency-dependence of calmodulin activation in dendritic spines. Neurocomputing. 2001;38-40:9–16. doi:https://doi.org/10.1016/S0925-2312(01)00415-5.

76. Keller DX, Franks KM, Bartol TM Jr, Sejnowski TJ. Calmodulin activation by calcium transients in the postsynaptic density of dendritic spines. PLoS ONE. 2008;3(4):e2045. doi:10.1371/journal.pone.0002045.

77. Weinan E, Lu J. Multiscale modeling. Scholarpedia. 2011;6(10):11527.

78. Sterratt DC, Sorokina O, Armstrong JD. Integration of Rule-Based Models and Compartmental Models of Neurons. In: Maler O, Halász Á, Dang T, Piazza C, editors. Hybrid Systems Biology: Second International Workshop, HSB 2013, Taormina, Italy, September 2, 2013 and Third International Workshop, HSB 2014, Vienna, Austria, July 23-24, 2014, Revised Selected Papers. vol. 7699 of LNBI. Cham: Springer International Publishing; 2015. p. 143–158. Available from: `http://dx.doi.org/10.1007/978-3-319-27656-4_9`.

79. Lisman J. A mechanism for the Hebb and the anti-Hebb processes underlying learning and memory. Proc Natl Acad Sci USA. 1989;86(23):9574–9578.

80. Bhalla US, Iyengar R. Emergent Properties of Networks of Biological Signalling Pathways. Science. 1999;283:381–387.

81. Ajay SM, Bhalla US. A role for ERKII in synaptic pattern selectivity on the time-scale of minutes. Eur J Neurosci. 2004;20(10):2671–2680. doi:10.1111/j.1460-9568.2004.03725.x.

82. Castellani GC, Quinlan EM, Bersani F, Cooper LN, Shouval HZ. A model of bidirectional synaptic plasticity: from signaling network to channel conductance. Learning and MemoryLearn Memoryhttp://wwwlearnmemorg/. 2005;12(4):423–432.

83. Zhabotinsky AM, Camp RN, Epstein IR, Lisman JE. Role of the Neurogranin Concentrated in Spines in the Induction of Long-Term Potentiation. J Neurosci. 2006;26(28):7337–7347. doi:10.1523/jneurosci.0729-06.2006.

84. Urakubo H, Honda M, Froemke RC, Kuroda S. Requirement of an allosteric kinetics of NMDA receptors for spike timing-dependent plasticity. J Neurosci. 2008;28(13):3310–3323. doi:10.1523/jneurosci.0303-08.2008.

85. D'Alcantara P, Schiffmann SN, Swillens S. Bidirectional synaptic plasticity as a consequence of interdependent Ca2+-controlled phosphorylation and dephosphorylation pathways. Eur J Neurosci. 2003;17(12):2521–2528.

86. Bear MF. Bidirectional synaptic plasticity: from theory to reality. Philos Trans R Soc Lond, B, Biol Sci. 2003;358(1432):649–655.

87. Barria A, Muller D, Derkach V, Griffith LC, Soderling TR. Regulatory phosphorylation of AMPA-type glutamate receptors by CaM-KII during long-term potentiation. Science. 1997;276(5321):2042–2045.

88. Lee HK, Barbarosie M, Kameyama K, Bear MF, Huganir RL. Regulation of distinct AMPA receptor phosphorylation sites during bidirectional synaptic plasticity. Nature. 2000;405(6789):955–959. doi:10.1038/35016089.

89. Lee HK, Kameyama K, Huganir RL, Bear MF. NMDA Induces Long-Term Synaptic Depression and Dephosphorylation of the GluR1 Subunit of AMPA Receptors in Hippocampus. Neuron. 1998;21(5):1151–1162. doi:10.1016/s0896-6273(00)80632-7.

90. Castellani GC, Quinlan EM, Cooper LN, Shouval HZ. A biophysical model of bidirectional synaptic plasticity: Dependence of AMPA and NMDA receptors. Proc Natl Acad Sci USA. 2001;98:12772–12777.

91. Bienenstock EL, Cooper LN, Munro PW. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. J Neurosci. 1982;2:32–48.

92. Lüscher C, Xia H, Beattie EC, Carroll RC, von Zastrow M, Malenka RC, et al. Role of AMPA Receptor Cycling in Synaptic Transmission and Plasticity. Neuron. 1999;24(3):649–658. doi:10.1016/s0896-6273(00)81119-8.

93. Choquet D, Triller A. The dynamic synapse. Neuron. 2013;80(3):691–703.

94. Opazo P, Choquet D. A three-step model for the synaptic recruitment of AMPA receptors. Mol Cell Neurosci. 2011;46(1):1–8.

95. Esteban JA, Shi SHH, Wilson C, Nuriya M, Huganir RL, Malinow R. PKA phosphorylation of AMPA receptor subunits controls synaptic trafficking underlying plasticity. Nat Neurosci. 2003;6(2):136–143. doi:10.1038/nn997.

96. Granger AJ, Nicoll RA. Expression mechanisms underlying long-term potentiation: a postsynaptic view, 10 years on. Philos Trans R Soc Lond, B, Biol Sci. 2014;369(1633):20130136+. doi:10.1098/rstb.2013.0136.

97. Migliore M, Hoffman DA, Magee JC, Johnston D. Role of an A-Type K^+ Conductance in the Back-Propagation of Action Potentials in the Dendrites of Hippocampal Pyramidal Neurons. J Comput Neurosci. 1999;7:5–15.

98. Poirazi P, Brannon T, Mel BW. Arithmetic of subthreshold synaptic summation in a model CA1 pyramidal cell. Neuron. 2003;37:977–987.

99. Frey U, Morris R. Synaptic tagging: implications for late maintenance of hippocampal long-term potentiation. Trends Neurosci. 1998;21:181–188.

100. Smolen P, Baxter DA, Byrne JH. A model of the roles of essential kinases in the induction and expression of late long-term potentiation. Biophys J. 2006;90(8):2760–2775. doi:10.1529/biophysj.105.072470.

101. Tsokas P, Hsieh C, Yao Y, Lesburguères E, Wallace EJ, Tcherepanov A, et al. Compensation for PKMζ in long-term potentiation and spatial long-term memory in mutant mice. ELife. 2016;5. doi:10.7554/eLife.14846.

102. Cerovic M, D'Isa R, Tonini R, Brambilla R. Molecular and cellular mechanisms of dopamine-mediated behavioral plasticity in the striatum. Neurobiol Learn Mem. 2013;105:63–80. doi:10.1016/j.nlm.2013.06.013.

103. Beninger RJ, Gerdjikov TV. Dopamine-Glutamate Interactions in Reward-Related Incentive Learning. In: Dopamine and Glutamate in Psychiatric Disorders. Totowa, NJ: Humana Press; 2005. p. 319–354. Available from: `http://link.springer.com/10.1007/978-1-59259-852-6{_}14`.

104. Yger M, Girault JA. DARPP-32, Jack of All Trades... Master of Which? Front Behav Neurosci. 2011;5(September):56. doi:10.3389/fnbeh.2011.00056.

105. Lindskog M, Kim M, Wikström MA, Blackwell KT, Kotaleski JH. Transient calcium and dopamine increase PKA activity and DARPP-32 phosphorylation. PLoS Comput Biol. 2006;2(9):e119. doi:10.1371/journal.pcbi.0020119.

106. Barbano PE, Spivak M, Flajolet M, Nairn AC, Greengard P, Greengard L. A mathematical tool for exploring the dynamics of biological networks. Proc Natl Acad Sci USA. 2007;104(49):19169–19174. doi:10.1073/pnas.0709955104.

107. Valjent E, Pascoli V, Svenningsson P, Paul S, Enslen H, Corvol JC, et al. Regulation of a protein phosphatase cascade allows convergent dopamine and glutamate signals to activate ERK in the striatum. Proc Natl Acad Sci USA. 2005;102(2):491–6. doi:10.1073/pnas.0408305102.

108. Devroyea C, Cathala A, Maitrea M, Piazzaa PV, Abrousa DN, Revesta JM, et al. Serotonin2C receptor stimulation inhibits cocaine-induced Fos expression and DARPP-32 phosphorylation in the rat striatum independently of dopamine outflow. Neuropharmacology. 2015;89:375–381. doi:http://dx.doi.org/10.1016/j.neuropharm.2014.10.016.

109. Hara M, Fukui R, Hieda E, Kuroiwa M, Bateup HS, Kano T, et al. Role of adrenoceptors in the regulation of dopamine/DARPP-32 signaling in neostriatal neurons. J Neurochem. 2010;113(4):1046–59. doi:10.1111/j.1471-4159.2010.06668.x.

110. D'Angelo E. The organization of plasticity in the cerebellar cortex: from synapses to control. Prog Brain Res. 2014;210:31–58. doi:10.1016/B978-0-444-63356-9.00002-9.

111. Kuroda S, Schweighofer N, Kawato M. Exploration of signal transduction pathways in cerebellar long-term depression by kinetic simulation. J Neurosci. 2001;21(15):5693–702.

112. Antunes G, De Schutter E. A stochastic signaling network mediates the probabilistic induction of cerebellar long-term depression. J Neurosci. 2012;32(27):9288–300. doi:10.1523/JNEUROSCI.5976-11.2012.

113. Hines ML, Morse T, Migliore M, Carnevale NT, Shepherd GM. ModelDB: A Database to Support Computational Neuroscience. J Comput Neurosci. 2004;17(1):7–11. doi:10.1023/B:JCNS.0000023869.22017.2e.

114. Chelliah V, Juty N, Ajmera I, Ali R, Dumousseau M, Glont M, et al. BioModels: ten-year anniversary. Nucleic Acids Res. 2015;43(Database issue):D542–548. doi:10.1093/nar/gku1181.

115. Sivakumaran S, Hariharaputran S, Mishra J, Bhalla US. The Database of Quantitative Cellular Signaling: management and analysis of chemical kinetic models of signaling networks. Bionformatics. 2003;19(3):408–15.

116. Lloyd CM, Lawson JR, Hunter PJ, Nielsen PF. The CellML Model Repository. Bionformatics. 2008;24(18):2122–3. doi:10.1093/bioinformatics/btn390.

117. Li C, Donizelli M, Rodriguez N, Dharuri H, Endler L, Chelliah V, et al. BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. BMC Syst Biol. 2010;4:92. doi:10.1186/1752-0509-4-92.

118. Le Novère N, Finney A, Hucka M, Bhalla US, Campagne F, Collado-Vides J, et al. Minimum information requested in the annotation of biochemical models (MIRIAM). Nat Biotechnol. 2005;23(12):1509–15. doi:10.1038/nbt1156.

119. Kötter R. Postsynaptic integration of glutamatergic and dopaminergic signals in the striatum. Prog Neurobiol. 1994;44(2):163–196.

120. Hernandez AI, Blace N, Crary JF, Serrano PA, Leitges M, Libien JM, et al. Protein kinase Mζ synthesis from a brain mRNA encoding an independent protein kinase Cζ catalytic domain: implications for the molecular mechanism of memory. J Biol Chem. 2003;278(41):40305–40316. doi:10.1074/jbc.M307065200.

121. Maglott D, Ostell J, Pruitt KD, Tatusova T. Entrez Gene: gene-centered information at NCBI. Nucleic Acids Res. 2011;39(Database issue):D52–57. doi:10.1093/nar/gkq1237.

122. Whittaker V, Michaelson I, Kirkland RJA. The separation of synaptic vesicles from nerve-ending particles (synaptosomes). Biochem J. 1964;90(2):293.

123. Bai F, Weizmann FA. Synaptosome proteomics. In: Subcellular Proteomics. Springer; 2007. p. 77–98.

124. Vastagh C, Rodolosse A, Solymosi N, Liposits Z. Altered expression of genes encoding neurotransmitter receptors in GnRH neurons of proestrous mice. Front Cell Neurosci. 2016;10.

125. Silverman AJ, Hou-Yu A, Chen WP. Corticotropin-releasing factor synapses within the paraventricular nucleus of the hypothalamus. Neuroendocrinology. 1989;49(3):291–299.

126. Mystek P, Tworzydło M, Dziedzicka-Wasylewska M, Polit A. New insights into the model of dopamine D1 receptor and G-proteins interactions. BBA-Mol Cell Res 2015;1853(3):594–603.

127. Ahn JH, Sung JY, McAvoy T, Nishi A, Janssens V, Goris J, et al. The B"/PR72 subunit mediates Ca^{2+}-dependent dephosphorylation of DARPP-32 by protein phosphatase 2A. Proc Natl Acad Sci USA. 2007;104(23):9876–81. doi:10.1073/pnas.0703589104.

128. The Gene Ontology Consortium. Gene Ontology Consortium: going forward. Nucleic Acids Res. 2015;43(D1):D1049. doi:10.1093/nar/gku1179.

129. Fabregat A, Sidiropoulos K, Garapati P, Gillespie M, Hausmann K, Haw R, et al. The Reactome pathway Knowledgebase. Nucleic Acids Res. 2016;44(D1):D481. doi:10.1093/nar/gkv1351.

130. Kibbe WA, Arze C, Felix V, Mitraka E, Bolton E, Fu G, et al. Disease Ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data. Nucleic Acids Res. 2015;43(D1):D1071. doi:10.1093/nar/gku1011.

131. Jimeno-Yepes AJ, Sticco JC, Mork JG, Aronson AR. GeneRIF indexing: sentence selection based on machine learning. BMC Bioinformatics. 2013;14(1):171.

132. McKusick VA. Mendelian inheritance in man: a catalog of human genes and genetic disorders. vol. 1. JHU Press; 1998.

133. Amberger J, Bocchini CA, Scott AF, Hamosh A. McKusick's online Mendelian inheritance in man (OMIM®). Nucleic Acids Res. 2009;37(suppl 1):D793–D796.

134. Chen Y, Cunningham F, Rios D, McLaren WM, Smith J, Pritchard B, et al. Ensembl variation resources. BMC Genomics. 2010;11(1):293.

135. He X, Simpson TI. statbio/topOnto: topOnto v1.0; 2017. Available from: `https://doi.org/10.5281/zenodo.819735`.

136. He X, Simpson TI. statbio/OntoSuite-Miner: OntoSuite-Miner v1.0; 2017. Available from: `https://doi.org/10.5281/zenodo.819726`.

137. Qi Z, Miller GW, Voit EO. Computational systems analysis of dopamine metabolism. PLoS ONE. 2008;3(6):e2444.

138. Sass MB, Lorenz AN, Green RL, Coleman RA. A pragmatic approach to biochemical systems theory applied to an α-synuclein-based model of Parkinson's disease. J Neurosci Methods. 2009;178(2):366–377.

139. Harrison PJ, Weinberger DR. Schizophrenia genes, gene expression, and neuropathology: on the matter of their convergence. Mol Psychiatr. 2005;10(1):40.

140. Männistö PT, Kaakkola S. Catechol-O-methyltransferase (COMT): biochemistry, molecular biology, pharmacology, and clinical efficacy of the new selective COMT inhibitors. Pharmacol Rev. 1999;51(4):593–628.

141. Weinshilboum RM, Otterness DM, Szumlanski CL. Methylation pharmacogenetics: catechol O-methyltransferase, thiopurine methyltransferase, and histamine N-methyltransferase. Annu Rev Pharmacol. 1999;39(1):19–52.

142. De Araujo ME, Erhart G, Buck K, Müller-Holzner E, Hubalek M, Fiegl H, et al. Polymorphisms in the gene regions of the adaptor complex LAMTOR2/LAMTOR3 and their association with breast cancer risk. PLoS ONE. 2013;8(1):e53768.

143. Sterratt D, Graham B, Gillies A, Willshaw D. Principles of Computational Modelling in Neuroscience. Cambridge, UK: Cambridge University Press; 2011.

144. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol (Lond). 1952;117:500–544.

145. Gray KA, Seal RL, Tweedie S, Wright MW, Bruford EA. A review of the new HGNC gene family resource. Hum Genomics. 2016;10:6. doi:10.1186/s40246-016-0062-6.

146. Larson SD, Martone ME. NeuroLex.org: an online framework for neuroscience knowledge. Front Neuroinform. 2013;7:18. doi:10.3389/fninf.2013.00018.

147. Apweiler R, Attwood TK, Bairoch A, Bateman A, Birney E, Biswas M, et al. The InterPro database, an integrated documentation resource for protein families, domains and functional sites. Nucleic Acids Res. 2001;29(1):37–40. doi:10.1093/nar/29.1.37.

148. Sheridan RP, Venkataraghavan R. A systematic search for protein signature sequences. Proteins. 1992;14(1):16–28. doi:10.1002/prot.340140105.

149. Orengo CA, Bateman A, Uversky V, editors. Protein families: relating protein sequence, structure, and function. Wiley; 2014.

150. Rivals I, Personnaz L, Taing L, Potier MC. Enrichment or depletion of a GO category within a class of genes: which test? Bionformatics. 2007;23(4):401–7. doi:10.1093/bioinformatics/btl633.

151. Benjamini Y, Yekutieli D. The control of the false discovery rate in multiple testing under depencency. Ann Stat. 2001;29(4):1165–1188. doi:10.1214/aos/1013699998.

152. Alexa A, Rahnenführer J, Lengauer T. Improved scoring of functional groups from gene expression data by decorrelating GO graph structure. Bionformatics. 2006;22(13):1600–1607. doi:10.1093/bioinformatics/btl140.

Bibliography

[1] B. Alberts. *Essential Cell Biology*. Garland Science, 2010.

[2] L. Bardwell, X. Zou, Q. Nie, and N. L. Komarova. Mathematical models of specificity in cell signaling. *Biophysical journal*, 92(10):3425–41, 2007.

[3] C. A. Orengo and A. Bateman, editors. *Protein Families: Relating Protein Sequence, Structure, and Function*. John Wiley & Sons, Inc., 2014.

[4] D. Ekman, Å. K. Björklund, J. Frey-Skött, and A. Elofsson. Multi-domain Proteins in the Three Kingdoms of Life: Orphan Domains and Other Unassigned Regions. *Journal of Molecular Biology*, 348(1):231–243, 2005.

[5] T. Pawson and P. Nash. Assembly of cell regulatory systems through protein interaction domains. *Science (New York, N.Y.)*, 300(5618):445–52, 2003.

[6] E. Bornberg-Bauer, F. Beaussart, S. K. Kummerfeld, S. a. Teichmann, and J. Weiner. The evolution of domain arrangements in proteins and interaction networks. *Cellular and molecular life sciences : CMLS*, 62(4):435–45, 2005.

[7] B. Schuster-Böckler and A. Bateman. Reuse of structural domain-domain interactions in protein networks. *BMC bioinformatics*, 8:259, 2007.

[8] S. Pasek, J.-L. Risler, and P. Brezellec. Gene fusion/fission is a major contributor to evolution of multi-domain bacterial proteins. *Bioinformatics*, 22(12):1418–1423, 2006.

[9] L. Mollica, L. M. Bessa, X. Hanoulle, M. R. Jensen, M. Blackledge, and R. Schneider. Binding Mechanisms of Intrinsically Disordered Proteins: Theory, Simulation, and Experiment. *Frontiers in molecular biosciences*, 3:52, 2016.

[10] K. Van Roey, B. Uyar, R. J. Weatheritt, H. Dinkel, M. Seiler, A. Budd, T. J. Gibson, and N. E. Davey. Short linear motifs: Ubiquitous and functionally diverse protein interaction modules directing cell regulation. *Chemical Reviews*, 114(13):6733–6778, 2014.

[11] K. N. Pandey. Functional roles of short sequence motifs in the endocytosis of membrane receptors. *Frontiers in Bioscience*, 14(1):5339, 2009.

[12] F. Diella, N. Haslam, C. Chica, A. Budd, S. Michael, N. P. Brown, G. Trave, and T. J. Gibson. Understanding eukaryotic linear motifs and their role in cell signaling and regulation. *Frontiers in bioscience : a journal and virtual library*, 13:6580–603, 2008.

[13] C. T. Walsh, S. Garneau-tsodikova, and G. J. Gatto. Protein posttranslational modifications: the chemistry of proteome diversifications. *Angewandte Chemie*, 44 45:7342–72, 2005.

[14] M. Mumby and D. Brekken. Phosphoproteomics: new insights into cellular signaling. *Genome biology*, 6(9):230, 2005.

[15] C. M. Stultz, A. D. Levin, and E. R. Edelman. Phosphorylation-induced Conformational Changes in a Mitogen-activated Protein Kinase Substrate. *Journal of Biological Chemistry*, 277(49):47653–47661, 2002.

[16] Y. Liu and M. R. Chance. Integrating phosphoproteomics in systems biology. *Computational and structural biotechnology journal*, 10(17):90–7, 2014.

[17] A. Nishi, M. Matamales, V. Musante, E. Valjent, M. Kuroiwa, Y. Kitahara, H. Rebholz, P. Greengard, J.-A. Girault, and A. C. Nairn. Glutamate Counteracts Dopamine/PKA Signaling via Dephosphorylation of DARPP-32 Ser-97 and Alteration of Its Cytonuclear Distribution. *The Journal of biological chemistry*, 292(4):1462–1476, 2017.

[18] J. Chen, F. Zeng, S. J. Forrester, S. Eguchi, M.-Z. Zhang, and R. C. Harris. Expression and Function of the Epidermal Growth Factor Receptor in Physiology and Disease. *Physiological Reviews*, 96(3):1025–1069, 2016.

[19] D. R. Robinson, Y.-M. Wu, and S.-F. Lin. The protein tyrosine kinase family of the human genome. *Oncogene*, 19(49):5548–5557, 2000.

[20] M. C. Good, J. G. Zalatan, and W. A. Lim. Scaffold proteins: hubs for controlling the flow of cellular information. *Science (New York, N.Y.)*, 332(6030):680–6, 2011.

[21] A. Zeke, M. Lukács, W. A. Lim, and A. Reményi. Scaffolds: interaction platforms for cellular signalling circuits. *Trends in cell biology*, 19(8):364–74, 2009.

[22] A. Peselis, A. Gao, and A. Serganov. Cooperativity, allostery and synergism in ligand binding to riboswitches. *Biochimie*, 117:100–9, 2015.

[23] M. I. Stefan and N. Le Novère. Cooperative Binding. *PLoS Computational Biology*, 9(6): e1003106, 2013.

[24] S. Y. Kim and J. E. Ferrell. Substrate Competition as a Source of Ultrasensitivity in the Inactivation of Wee1. *Cell*, 128(6):1133–1145, 2007.

[25] M. A. Rowland, W. Fontana, and E. J. Deeds. Crosstalk and competition in signaling networks. *Biophysical journal*, 103(11):2389–98, 2012.

[26] C. Y. Huang and J. E. Ferrell. Ultrasensitivity in the mitogen-activated protein kinase cascade. *Proceedings of the National Academy of Sciences*, 93(19):10078–10083, 1996.

[27] U. S. Bhalla and R. Iyengar. Robustness of the bistable behavior of a biological signaling feedback loop. *Chaos (Woodbury, N.Y.)*, 11(1):221–226, 2001.

[28] P. Csermely, T. Korcsmáros, H. J. M. Kiss, G. London, and R. Nussinov. Structure and dynamics of molecular networks: a novel paradigm of drug discovery: a comprehensive review. *Pharmacology & therapeutics*, 138(3):333–408, 2013.

[29] J. J. Tyson, K. C. Chen, and B. Novak. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Current Opinion in Cell Biology*, 15(2): 221–231, 2003.

[30] B. N. Kholodenko. Cell-signalling dynamics in time and space. *Nature reviews. Molecular cell biology*, 7(3):165–76, 2006.

[31] U. Alon. Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461, 2007.

[32] U. Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman & Hall/CRC Mathematical and Computational Biology. Taylor & Francis, 2006.

[33] S. Mangan and U. Alon. Structure and function of the feed-forward loop network motif. *Proceedings of the National Academy of Sciences*, 100(21):11980–11985, 2003.

[34] Q. Zhang, S. Bhattacharya, R. B. Conolly, H. J. C. Iii, N. E. Kaminski, and M. E. Andersen. Molecular Signaling Network Motifs Provide a Mechanistic Basis for Cellular Threshold Responses. *Environmental Health Perspectives*, 122(12):1261–1270, 2014.

[35] E. Kandel, T. Jessell, J. Schwartz, S. Siegelbaum, and A. Hudspeth. *Principles of Neural Science, Fifth Edition*. Principles of Neural Science. McGraw-Hill Education, 2013.

[36] T. Takeuchi, A. J. Duszkiewicz, and R. G. M. Morris. The synaptic plasticity and memory hypothesis: encoding, storage and persistence. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 369(1633):20130288, 2014.

[37] A. Citri and R. C. Malenka. Synaptic Plasticity: Multiple Forms, Functions and Mechanisms. *Neuropsychopharmacology*, 33(1):18–41, 2008.

[38] K. T. Blackwell and J. Jedrzejewska-Szmek. Molecular mechanisms underlying neuronal synaptic plasticity: Systems biology meets computational neuroscience in the wilds of synaptic plasticity. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 5(6): 717–731, 2013.

[39] D. J. Surmeier, J. Plotkin, and W. Shen. Dopamine and synaptic plasticity in dorsal striatal circuits controlling action selection. *Current Opinion in Neurobiology*, 19(6):621–628, 2009.

[40] C. Lüscher, H. Xia, E. C. Beattie, R. C. Carroll, M. von Zastrow, R. C. Malenka, and R. A. Nicoll. Role of AMPA Receptor Cycling in Synaptic Transmission and Plasticity. *Neuron*, 24(3):649–658, 1999.

[41] Y. Goto, C. R. Yang, and S. Otani. Functional and dysfunctional synaptic plasticity in prefrontal cortex: roles in psychiatric disorders. *Biological psychiatry*, 67(3):199–207, 2010.

[42] T. V. P. Bliss, G. L. Collingridge, and R. G. M. Morris. Synaptic plasticity in health and disease: introduction and overview. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 369(1633):20130129, 2014.

[43] G. W. Crabtree and J. A. Gogos. Synaptic plasticity, neural circuits, and the emerging role of altered short-term information processing in schizophrenia. *Frontiers in Synaptic Neuroscience*, 6:28, 2014.

[44] T. Kimura, D. J. Whitcomb, J. Jo, P. Regan, T. Piers, S. Heo, C. Brown, T. Hashikawa, M. Murayama, H. Seok, I. Sotiropoulos, E. Kim, G. L. Collingridge, A. Takashima, and K. Cho. Microtubule-associated protein tau is essential for long-term depression in the hippocampus. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369 (1633):20130144–20130144, 2013.

[45] K. F. Heil, E. Wysocka, O. Sorokina, J. H. Kotaleski, T. I. Simpson, J. D. Armstrong, and D. C. Sterratt. Analysis of proteins in computational models of synaptic plasticity. *bioRxiv*, 2018.

[46] S. Gal-Ben-Ari, J. W. Kenney, H. Ounalla-Saad, E. Taha, O. David, D. Levitan, I. Gildish, D. Panja, B. Pai, K. Wibrand, T. I. Simpson, C. G. Proud, C. R. Bramham, J. D. Armstrong, and K. Rosenblum. Consolidation and translation regulation. *Learning & memory (Cold Spring Harbor, N.Y.)*, 19(9):410–22, 2012.

[47] N. X. Tritsch and B. L. Sabatini. Dopaminergic modulation of synaptic transmission in

cortex and striatum. *Neuron*, 76(1):33–50, 2012.

[48] H. Kitano. Systems Biology: A Brief Overview. *Science*, 295(5560), 2002.

[49] U. S. Bhalla and R. Iyengar. Emergent properties of networks of biological signaling pathways. *Science*, 283(5400):381–387, 1999.

[50] E. A. Sobie, Y.-S. Lee, S. L. Jenkins, and R. Iyengar. Systems biology–biomedical modeling. *Science signaling*, 4(190):tr2, 2011.

[51] Y. Hasin, M. Seldin, and A. Lusis. Multi-omics approaches to disease. *Genome Biology*, 18(1):83, 2017.

[52] B. M. Neale, J. Lasky-Su, R. Anney, B. Franke, K. Zhou, J. B. Maller, A. A. Vasquez, P. Asherson, W. Chen, T. Banaschewski, J. Buitelaar, R. Ebstein, M. Gill, A. Miranda, R. D. Oades, H. Roeyers, A. Rothenberger, J. Sergeant, H. C. Steinhausen, E. Sonuga-Barke, F. Mulas, E. Taylor, N. Laird, C. Lange, M. Daly, and S. V. Faraone. Genome-wide association scan of attention deficit hyperactivity disorder. *American journal of medical genetics. Part B, Neuropsychiatric genetics : the official publication of the International Society of Psychiatric Genetics*, 147B(8):1337–44, 2008.

[53] T. B. Schreiber, N. Mäusbacher, S. B. Breitkopf, K. Grundner-Culemann, and H. Daub. Quantitative phosphoproteomics–an emerging key technology in signal-transduction research. *Proteomics*, 8(21):4416–32, 2008.

[54] D. J. McGrail, L. Federico, Y. Li, H. Dai, Y. Lu, G. B. Mills, S. Yi, S.-Y. Lin, and N. Sahni. Multi-omics analysis reveals neoantigen-independent immune cell infiltration in copy-number driven cancers. *Nature Communications*, 9(1):1317, 2018.

[55] P. Khatri, M. Sirota, and A. J. Butte. Ten Years of Pathway Analysis: Current Approaches and Outstanding Challenges. *PLoS Computational Biology*, 8(2):e1002375, 2012.

[56] T. Ideker and R. Sharan. Protein networks in disease. *Genome research*, 18(4):644–52, 2008.

[57] L. Xie, X. Ge, H. Tan, L. Xie, Y. Zhang, T. Hart, X. Yang, and P. E. Bourne. Towards Structural Systems Pharmacology to Study Complex Diseases and Personalized Medicine. *PLoS Computational Biology*, 10(5):e1003554, 2014.

[58] W. Wiechert and S. Noack. Mechanistic pathway modeling for industrial biotechnology: challenging but worthwhile. *Current Opinion in Biotechnology*, 22(5):604–610, 2011.

[59] D. Cook, D. Brown, R. Alexander, R. March, P. Morgan, G. Satterthwaite, and M. N. Pangalos. Lessons learned from the fate of AstraZeneca's drug pipeline: a five-dimensional framework. *Nature Reviews Drug Discovery*, 13(6):419–431, 2014.

[60] G. Bebek, M. Koyutürk, N. D. Price, and M. R. Chance. Network biology methods integrating biological data for translational science. *Briefings in bioinformatics*, 13(4): 446–59, 2012.

[61] X. Wang, N. Gulbahce, and H. Yu. Network-based methods for human disease gene prediction. *Briefings in functional genomics*, 10(5):280–93, 2011.

[62] A. Ma'ayan, A. D. Rouillard, N. R. Clark, Z. Wang, Q. Duan, and Y. Kou. Lean Big Data integration in systems biology and systems pharmacology. *Trends in pharmacological sciences*, 35(9):450–60, 2014.

[63] B. H. Junker and F. Schreiber. *Analysis of Biological Networks (Wiley Series in Bioinformatics)*. Wiley-Interscience, 2008.

[64] A.-L. Barabási, N. Gulbahce, and J. Loscalzo. Network medicine: a network-based approach to human disease. *Nature reviews. Genetics*, 12(1):56–68, 2011.

[65] S. Wuchty and E. Almaas. Peeling the yeast protein network. *PROTEOMICS*, 5(2): 444–449, 2005.

[66] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, 2001.

[67] X. Zhang, T. Martin, and M. E. Newman. Identification of core-periphery structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 2015.

[68] M. Caldera, P. Buphamalai, F. Müller, and J. Menche. Interactome-based approaches to human disease. *Current Opinion in Systems Biology*, 3:88–94, 2017.

[69] H. W. Han, J. H. Ohn, J. Moon, and J. H. Kim. Yin and Yang of disease genes and death genes between reciprocally scale-free biological networks. *Nucleic acids research*, 41(20): 9209–17, 2013.

[70] R. A. Hanneman and M. Riddle. *Introduction to social network methods*. University of California, Riverside, Riverside, CA, 2005.

[71] L. Yang, X. Zhao, and X. Tang. Predicting disease-related proteins based on clique backbone in protein-protein interaction network. *International journal of biological sciences*, 10(7):677–88, 2014.

[72] V. Spirin and L. A. Mirny. Protein complexes and functional modules in molecular networks. *Proceedings of the National Academy of Sciences of the United States of America*, 100(21):12123–8, 2003.

[73] V. Batagelj and M. Zaversnik. An O(m) Algorithm for Cores Decomposition of Networks. *arXiv*, 2003.

[74] J. I. Alvarez-Hamelin, L. Dall'Asta, A. Barrat, and A. Vespignani. K-Core Decomposition: a Tool for the Visualization of Large Scale Networks. *arXiv*, 2005.

[75] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[76] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12):7821–6, 2002.

[77] V. Estivill-Castro. Why So Many Clustering Algorithms: A Position Paper. *SIGKDD Explor. Newsl.*, 4(1):65–75, 2002.

[78] M. A. Javed, M. S. Younis, S. Latif, J. Qadir, and A. Baig. Community detection in networks: A multidisciplinary review. *Journal of Network and Computer Applications*, 108: 87–111, 2018.

[79] Z. Yang, R. Algesheimer, and C. J. Tessone. A Comparative Analysis of Community Detection Algorithms on Artificial Networks. *Scientific Reports*, 6(1):30750, 2016.

[80] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 69(6 2):1–5, 2004.

[81] S. Emmons, S. Kobourov, M. Gallant, and K. Börner. Analysis of Network Clustering Algorithms and Cluster Quality Metrics at Scale. *PLOS ONE*, 11(7):e0159161, 2016.

[82] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of commu-

nities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10): P10008, 2008.

[83] J. H. Malone and B. Oliver. Microarrays, deep sequencing and the true measure of the transcriptome. *BMC Biology*, 9(1):34, 2011.

[84] M. Schena and D. Shalon. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270(5235):467–470, 1995.

[85] C.-A. Tsai, Y.-J. Chen, and J. J. Chen. Testing for differentially expressed genes with microarray data. *Nucleic acids research*, 31(9):e52, 2003.

[86] L. Zhang, S. Chang, Z. Li, K. Zhang, Y. Du, J. Ott, and J. Wang. ADHDgene: a genetic database for attention deficit hyperactivity disorder. *Nucleic acids research*, 40(Database issue):D1003–9, 2012.

[87] I. Rivals, L. Personnaz, L. Taing, and M.-C. Potier. Enrichment or depletion of a GO category within a class of genes: which test? *Bioinformatics (Oxford, England)*, 23(4): 401–7, 2007.

[88] M. Uschold, M. Gruninger, M. Uschold, and M. Gruninger. Ontologies : Principles , Methods and Applications. *Knowledge Engineering Review*, 11(2):93–136, 1996.

[89] M. A. Musen, N. F. Noy, N. H. Shah, P. L. Whetzel, C. G. Chute, M.-A. Story, B. Smith, and N. team. The National Center for Biomedical Ontology. *Journal of the American Medical Informatics Association : JAMIA*, 19(2):190–5, 2012.

[90] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[91] The Gene Ontology Consortium. Expansion of the gene ontology knowledgebase and resources. *Nucleic Acids Research*, 45(D1):D331–D338, 2017.

[92] S. Chowdhury and R. R. Sarkar. Comparison of human cell signaling pathway databases–evolution, drawbacks and challenges. *Database*, 2015(0):bau126–bau126, 2015.

[93] M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res*, 28(1):27–30, 2000.

[94] D. Croft, G. O'Kelly, G. Wu, R. Haw, M. Gillespie, L. Matthews, M. Caudy, P. Garapati, G. Gopinath, B. Jassal, S. Jupe, I. Kalatskaya, S. Mahajan, B. May, N. Ndegwa, E. Schmidt, V. Shamovsky, C. Yung, E. Birney, H. Hermjakob, P. D'Eustachio, and L. Stein. Reactome: a database of reactions, pathways and biological processes. *Nucleic acids research*, 39 (Database issue):D691–7, 2011.

[95] E. G. Cerami, B. E. Gross, E. Demir, I. Rodchenkov, O. Babur, N. Anwar, N. Schultz, G. D. Bader, and C. Sander. Pathway Commons, a web resource for biological pathway data. *Nucleic acids research*, 39(Database issue):D685–90, 2011.

[96] A. R. Pico, T. Kelder, M. P. van Iersel, K. Hanspers, B. R. Conklin, and C. Evelo. WikiPathways: Pathway Editing for the People. *PLoS Biology*, 6(7):e184, 2008.

[97] M. Hucka, F. T. Bergmann, S. M. Keating, J. C. Schaff, L. P. Smith, D. J. Wilkinson, S. Hoops, S. M. Keating, S. Sahle, J. C. Schaff, L. P. Smith, and D. J. Wilkinson. The Systems Biology Markup Language (SBML): Language Specification for Level 3 Version

1 Core. *Nature Precedings*, 2010.

[98] E. Demir, M. P. Cary, S. Paley, K. Fukuda, C. Lemer, G. Wu, P. D. Eustachio, C. Schaefer, and J. Luciano. BioPAX – A community standard for pathway data sharing. *Nature Biotechnology*, 28(9):935–942, 2011.

[99] N. Yu, J. Seo, K. Rho, Y. Jang, J. Park, W. K. Kim, and S. Lee. hiPathDB: a human-integrated pathway database with facile visualization. *Nucleic acids research*, 40(Database issue):D797–802, 2012.

[100] H. Zhou, J. Jin, H. Zhang, B. Yi, M. Wozniak, and L. Wong. IntPath–an integrated pathway gene relationship database for model organisms and important pathogens. *BMC systems biology*, 6 Suppl 2:S2, 2012.

[101] F. Belinky, N. Nativ, G. Stelzer, S. Zimmerman, T. Iny Stein, M. Safran, and D. Lancet. PathCards: multi-source consolidation of human biological pathways. *Database : the journal of biological databases and curation*, 2015, 2015.

[102] A. Kamburov, U. Stelzl, H. Lehrach, and R. Herwig. The ConsensusPathDB interaction database: 2013 update. *Nucleic acids research*, 41(Database issue):D793–800, 2013.

[103] D. C. Kirouac, J. Saez-Rodriguez, J. Swantek, J. M. Burke, D. A. Lauffenburger, and P. K. Sorger. Creating and analyzing pathway and protein interaction compendia for modelling signal transduction networks. *BMC Systems Biology*, 6(1):29, 2012.

[104] J. C. Sible and J. J. Tyson. Mathematical modeling as a tool for investigating cell cycle control networks. *Methods (San Diego, Calif.)*, 41(2):238–47, 2007.

[105] E. Klipp, R. Herwig, A. Kowald, C. Wierling, and H. Lehrach. *Systems Biology in Practice: Concepts, Implementation and Application*. Wiley-VCH, 2005.

[106] W. S. Hlavacek, J. R. Faeder, M. L. Blinov, R. G. Posner, M. Hucka, and W. Fontana. Rules for modeling signal-transduction systems. *Science's STKE : signal transduction knowledge environment*, 2006(344):re6, 2006.

[107] D. Wilkinson. *Stochastic Modelling for Systems Biology*. Chapman & Hall/CRC Mathematical & Computational Biology. Taylor & Francis, 2006.

[108] L. A. Chylek, L. A. Harris, J. R. Faeder, and W. S. Hlavacek. Modeling for (physical) biologists: an introduction to the rule-based approach. *Physical biology*, 12(4):045007, 2015.

[109] C. Knüpfer, C. Beckstein, P. Dittrich, and N. Le Novère. Structure, function, and behaviour of computational models in systems biology. *BMC systems biology*, 7:43, 2013.

[110] S. H. Strogatz. *Nonlinear dynamics and chaos : with applications to physics, biology, chemistry, and engineering*. Studies in nonlinearity. Westview Press, 1 edition, 2000.

[111] A. Dräger, N. Hassis, J. Supper, A. Schröder, and A. Zell. SBMLsqueezer: a CellDesigner plug-in to generate kinetic rate equations for biochemical networks. *BMC systems biology*, 2(1):39, 2008.

[112] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI–a COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074, 2006.

[113] D. J. Wilkinson. Stochastic modelling for quantitative description of heterogeneous biological systems. *Nature reviews. Genetics*, 10(2):122–33, 2009.

[114] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4):403–434, 1976.

[115] Y. Chen, F. Cunningham, D. Rios, W. M. McLaren, J. Smith, B. Pritchard, G. M. Spudich, S. Brent, E. Kulesha, P. Marin-Garcia, D. Smedley, E. Birney, and P. Flicek. Ensembl variation resources. *BMC Genomics*, 11(1):293, 2010.

[116] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain. Stochastic gene expression in a single cell. *Science (New York, N.Y.)*, 297(5584):1183–6, 2002.

[117] S. Kontogeorgaki, R. J. Sánchez-García, R. M. Ewing, K. C. Zygalakis, and B. D. MacArthur. Noise-processing by signaling networks. *Scientific Reports*, 7(1):532, 2017.

[118] J. E. Ladbury and S. T. Arold. Noise in cellular signaling pathways: causes and effects. *Trends in biochemical sciences*, 37(5):173–8, 2012.

[119] D. T. Gillespie. Exact Stochastic Simulation of Coupled Chemical Reactions. *Journal of Physical Chemistry*, 81(1):2340–2361, 1977.

[120] V. Danos. *Rule-Based Modelling of Cellular Signalling*, volume 4703. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[121] P. Seshacharyulu, M. P. Ponnusamy, D. Haridas, M. Jain, A. K. Ganti, and S. K. Batra. Targeting the EGFR signaling pathway in cancer therapy. *Expert opinion on therapeutic targets*, 16(1):15–31, 2012.

[122] B. J. Mayer, M. L. Blinov, and L. M. Loew. Molecular machines or pleiomorphic ensembles: signaling complexes revisited. *Journal of biology*, 8(9):81, 2009.

[123] R. Suderman and E. J. Deeds. Machines vs. Ensembles: Effective MAPK Signaling through Heterogeneous Sets of Protein Complexes. *PLoS Computational Biology*, 9(10): e1003278, 2013.

[124] L. A. Chylek, L. A. Harris, C.-S. Tung, J. R. Faeder, C. F. Lopez, and W. S. Hlavacek. Rule-based modeling: a computational approach for studying biomolecular site dynamics in cell signaling systems. *Wiley interdisciplinary reviews. Systems biology and medicine*, 6(1): 13–36, 2014.

[125] E. Bartocci and P. Lió. Computational Modeling, Formal Analysis, and Tools for Systems Biology. *PLoS Computational Biology*, 12(1):e1004591, 2016.

[126] D. Machado, R. S. Costa, M. Rocha, E. C. Ferreira, B. Tidor, and I. Rocha. Modeling formalisms in Systems Biology. *AMB Express, Springer Open Journal*, 1(45), 2011.

[127] N. Tenazinha and S. Vinga. A Survey on Methods for Modeling and Analyzing Integrated Biological Networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(4):943–958, 2011.

[128] Z. Ji, K. Yan, W. Li, H. Hu, and X. Zhu. Mathematical and Computational Modeling in Complex Biological Systems. *BioMed Research International*, 2017:1–16, 2017.

[129] N. Le Novère. Quantitative and logic modelling of molecular and gene networks. *Nature Reviews Genetics*, 16(3):146–158, 2015.

[130] J. R. Karr, J. C. Sanghvi, D. N. Macklin, M. V. Gutschow, J. M. Jacobs, B. Bolival, N. Assad-Garcia, J. I. Glass, and M. W. Covert. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389–401, 2012.

[131] M. Mattioni and N. Le Novère. Integration of Biochemical and Electrical Signaling-Multiscale Model of the Medium Spiny Neuron of the Striatum. *PLoS ONE*, 8(7):e66811, 2013.

[132] D. C. Sterratt, O. Sorokina, and J. D. Armstrong. Integration of rule-based models and compartmental models of neurons. In O. Maler, Á. Halász, T. Dang, and C. Piazza, editors, *Hybrid Systems Biology: Second International Workshop, HSB 2013, Taormina, Italy, September 2, 2013 and Third International Workshop, HSB 2014, Vienna, Austria, July 23-24, 2014, Revised Selected Papers*, volume 7699 of *LNBI*, pages 143–158. Springer International Publishing, Cham, 2015.

[133] O. Sorokina, A. Sorokin, J. D. Armstrong, and V. Danos. A simulator for spatially extended kappa models. *Bioinformatics*, 29(23):3105–3106, 2013.

[134] B. Hoard, B. Jacobson, K. Manavi, and L. Tapia. Extending rule-based methods to model molecular geometry and 3D model resolution. *BMC Systems Biology*, 10(48):121–138, 2016.

[135] J. Baeten. A brief history of process algebra. *Theoretical Computer Science*, 335(2-3): 131–146, 2005.

[136] F. Ciocchetta and J. Hillston. Process algebras in systems biology. In M. Bernardo, P. Degano, and G. Zavattaro, editors, *Formal Methods for Computational Systems Biology*, pages 265–312, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[137] A. Regev, W. Silverman, and E. Shapiro. Representation and simulation of biochemical processes using the pi-calculus process algebra. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pages 459–70, 2001.

[138] M. L. Guerriero, C. Priami, and A. Romanel. Modeling static biological compartments with beta-binders. In H. Anai, K. Horimoto, and T. Kutsia, editors, *Algebraic Biology*, pages 247–261, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[139] L. Dematté, R. Larcher, A. Palmisano, C. Priami, and A. Romanel. Programming Biology in BlenX. In S. Choi, editor, *Systems Biology for Signaling Networks*, chapter 31, pages 777–820. Springer-Verlag New York, 1 edition, 2010.

[140] M. Hucka, A. Finney, B. Bornstein, S. Keating, B. Shapiro, J. Matthews, B. Kovitz, M. Schilstra, A. Funahashi, J. Doyle, and H. Kitano. Evolving a lingua franca and associated software infrastructure for computational systems biology: the Systems Biology Markup Language (SBML) project. *Systems Biology*, 1(1):41–53, 2004.

[141] M. Hucka and L. P. Smith. SBML Level 3 package: Groups, Version 1 Release 1. *Journal of integrative bioinformatics*, 13(3):290, 2016.

[142] L. Cardelli, E. Caron, P. Gardner, O. Kahramanoğulları, and A. Phillips. A Process Model of Actin Polymerisation. *Electronic Notes in Theoretical Computer Science*, 229(1):127–144, 2009.

[143] R. Larcher, C. Priami, and A. Romanel. Modelling self-assembly in blenx. In C. Priami, R. Breitling, D. Gilbert, M. Heiner, and A. M. Uhrmacher, editors, *Transactions on Computational Systems Biology XII: Special Issue on Modeling Methodologies*, pages 163–198, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[144] F. Ciocchetta, M. L. Guerriero, and J. Hillston. Investigating modularity in the analysis

of process algebra models of biochemical systems. *Electronic Proceedings in Theoretical Computer Science*, 19:55–69, 2010.

[145] M. Pedersen, A. Phillips, and G. D. Plotkin. A high-level language for rule-based modelling. *PloS one*, 10(6):e0114296, 2015.

[146] F. Ciocchetta, A. Dugout, and M. L. Guerriero. A compartmental model of the cAMP/P-KA/MAPK pathway in Bio-PEPA. *Electronic Proceedings in Theoretical Computer Science*, 11:71–90, 2009.

[147] V. Danos and C. Laneve. Formal Molecular Biology. *Theoretical Computer Science*, 325: 69–110, 2004.

[148] E. Bonabeau. Agent-based modeling: methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences of the United States of America*, 99 Suppl 3(Suppl 3):7280–7, 2002.

[149] L. A. Chylek, E. C. Stites, R. G. Posner, and W. S. Hlavacek. Innovations of the Rule-Based Modeling Approach. In A. Prokop and B. Csukás, editors, *Systems Biology: Integrative Biology and Simulation Tools*, pages 273–300. Springer Netherlands, Dordrecht, 2013.

[150] M. I. Stefan, T. M. Bartol, T. J. Sejnowski, and M. B. Kennedy. Multi-state Modeling of Biomolecules. *PLoS Computational Biology*, 10(9):e1003844, 2014.

[151] J. R. Faeder, M. L. Blinov, and W. S. Hlavacek. Rule-Based Modeling of Biochemical Systems with BioNetGen. *Methods in Molecular Biology, Systems Biology*, 500:83–89, 2009.

[152] B. Liu and P. S. Thiagarajan. Modeling and analysis of biopathways dynamics. *Journal of bioinformatics and computational biology*, 10(4), 2012.

[153] C. F. Lopez, J. L. Muhlich, J. A. Bachman, and P. K. Sorger. Programming biological models in Python using PySB. *Molecular systems biology*, 9:646, 2013.

[154] J. A. P. Sekar and J. R. Faeder. Rule-based modeling of signal transduction: a primer. *Methods in molecular biology (Clifton, N.J.)*, 880:139–218, 2012.

[155] V. Danos, J. Feret, W. Fontana, R. Harmer, J. Hayman, J. Krivine, C. Thompson-Walsh, and G. Winskel. Graphs, Rewriting and Pathway Reconstruction for Rule-Based Models. *FSTTCS 2012 - IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, 18:276–288, 2012.

[156] F. Gadducci. Graph rewriting for the π-calculus. *Mathematical Structures in Computer Science*, 17(03):407, 2007.

[157] J. Feret and J. Krivine. KaSim3 reference manual, 2012.

[158] J. Feret, V. Danos, J. Krivine, R. Harmer, and W. Fontana. Internal coarse-graining of molecular systems. *Proceedings of the National Academy of Sciences of the United States of America*, 106(16):6453–8, 2009.

[159] J. Feret, V. Danos, J. Krivine, R. Harmer, and W. Fontana. Internal coarse-graining of molecular systems (Supporting Information). *Proceedings of the National Academy of Sciences of the United States of America*, 106(16):6453–8, 2009.

[160] M. Schon, G. Engeln-Müllges, F. Uhlig, and F. Uhlig. *Numerical Algorithms with C*. Springer Berlin Heidelberg, 2013.

[161] D. T. Gillespie. Stochastic simulation of chemical kinetics. *Annual review of physical chemistry*, 58:35–55, 2007.

[162] J. Yang and W. S. Hlavacek. The efficiency of reactant site sampling in network-free simulation of rule-based models for biochemical systems. *Physical biology*, 8(5):055009, 2011.

[163] V. Danos, J. Feret, W. Fontana, and J. Krivine. Scalable Simulation of Cellular Signaling Networks. In *Programming Languages and Systems*, pages 139–157. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[164] J. Krivine, V. Danos, and A. Benecke. Modelling Epigenetic Information Maintenance: A Kappa Tutorial. In A. Bouajjani and O. Maler, editors, *Computer Aided Verification*, pages 17–32, Berlin, Heidelberg, 2009. Springer-Verlag.

[165] M. W. Sneddon, J. R. Faeder, and T. Emonet. Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nature methods*, 8(2):177–183, 2011.

[166] E. J. Deeds, J. Krivine, J. Feret, V. Danos, and W. Fontana. Combinatorial complexity and compositional drift in protein interaction networks. *PloS one*, 7(3):e32032, 2012.

[167] P. M. Kim, L. J. Lu, Y. Xia, and M. B. Gerstein. Relating Three-Dimensional Structures to Protein Networks Provides Evolutionary Insights. *Science*, 314:1938–1941, 2006.

[168] O. Sorokina, A. Sorokin, and J. D. Armstrong. Towards a quantitative model of the post-synaptic proteome. *Molecular bioSystems*, 7(10):2813–2823, 2011.

[169] L. A. Chylek, B. S. Wilson, and W. S. Hlavacek. Modeling Biomolecular Site Dynamics in Immunoreceptor Signaling Systems. *Advances in Experimental Medicine and Biology*, 844: 245–262, 2014.

[170] G. Antunes, A. C. Roque, and F. M. Simoes-de Souza. Stochastic Induction of Long-Term Potentiation and Long-Term Depression. *Nature Scientific Reports*, 6(1):30899, 2016.

[171] B. Di Camillo, A. Carlon, F. Eduati, and G. M. Toffolo. A rule-based model of insulin signalling pathway. *BMC systems biology*, 10(1):38, 2016.

[172] J. A. Rohrs, P. Wang, and S. D. Finley. Predictive Model of Lymphocyte-Specific Protein Tyrosine Kinase (LCK) Autoregulation. *Cellular and molecular bioengineering*, 9:351–367, 2016.

[173] E. C. Stites, M. Aziz, M. S. Creamer, D. D. Von Hoff, R. G. Posner, and W. S. Hlavacek. Use of mechanistic models to integrate and analyze multiple proteomic datasets. *Biophysical journal*, 108(7):1819–29, 2015.

[174] A. Köhler, J. Krivine, and J. Vidmar. A Rule-Based Model of Base Excision Repair. In *CMSB 2014: Computational Methods in Systems Biology*, pages 173–195. Springer, Cham, 2014.

[175] L. A. Chylek, V. Akimov, J. Dengjel, K. T. G. Rigbolt, B. Hu, W. S. Hlavacek, and B. Blagoev. Phosphorylation site dynamics of early T-cell receptor signaling. *PloS one*, 9(8):e104240, 2014.

[176] J.-J. Tapia and J. R. Faeder. The Atomizer: Extracting Implicit Molecular Structure from Reaction Network Models. *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics (BCB'13)*, pages 726–727, 2013.

[177] É. Fernandez, R. Schiappa, J. A. Girault, and N. Le Novère. DARPP-32 is a robust integrator of dopamine and glutamate signals. *PLoS Computational Biology*, 2(12):1619–1633, 2006.

[178] T. Manninen, K. Hituri, E. Toivari, and M.-L. Linne. Modeling signal transduction leading to synaptic plasticity: evaluation and comparison of five models. *EURASIP journal on bioinformatics & systems biology*, 2011:797250, 2011.

[179] A. G. Nair, U. S. Bhalla, J. H. Kotaleski, and J. J. Saucerman. Role of DARPP32 and ARPP21 in the Emergence of Temporal Constraints on Striatal Calcium and Dopamine Integration. *PLOS Computational Biology PLoS Comput Biol*, 12(9), 2016.

[180] T. Nakano, T. Doi, J. Yoshimoto, and K. Doya. A kinetic model of dopamine- and calcium-dependent striatal synaptic plasticity. *PLoS Computational Biology*, 6(2):e1000670, 2010.

[181] J. W. Bales, H. Q. Yan, X. Ma, Y. Li, R. Samarasinghe, and C. E. Dixon. The dopamine and cAMP regulated phosphoprotein, 32kDa (DARPP-32) signaling pathway: A novel therapeutic target in traumatic brain injury. *Experimental Neurology*, 229(2):300–307, 2011.

[182] J. Kim, I. S. Ryu, S. Y. Seo, and E. S. Choe. Activation of Protein Kinases and Phosphatases Coupled to Glutamate Receptors Regulates the Phosphorylation State of DARPP32 at Threonine 75 After Repeated Exposure to Cocaine in the Rat Dorsal Striatum in a Ca2+-Dependent Manner. *The international journal of neuropsychopharmacology / official scientific journal of the Collegium Internationale Neuropsychopharmacologicum (CINP)*, 18(12):992–999, 2015.

[183] I. Buesa, Z. Aira, J. J. Azkue, J. Li, X. Zhang, and X. Chen. Regulation of Nociceptive Plasticity Threshold and DARPP-32 Phosphorylation in Spinal Dorsal Horn Neurons by Convergent Dopamine and Glutamate Inputs. *PLOS ONE*, 11(9):e0162416, 2016.

[184] N. Juty, R. Ali, M. Glont, S. Keating, N. Rodriguez, M. J. Swat, S. M. Wimalaratne, H. Hermjakob, N. Le Novère, C. Laibe, and V. Chelliah. BioModels: Content, Features, Functionality, and Use. *CPT: pharmacometrics & systems pharmacology*, 4(2):e3, 2015.

[185] N. L. Novère, A. Finney, M. Hucka, U. S. Bhalla, F. Campagne, J. Collado-Vides, E. J. Crampin, M. Halstead, E. Klipp, P. Mendes, P. Nielsen, H. Sauro, B. Shapiro, J. L. Snoep, H. D. Spence, and B. L. Wanner. Minimum information requested in the annotation of biochemical models (MIRIAM). *Nature Biotechnology*, 23(12):1509–1515, 2005.

[186] The UniProt Consortium. Uniprot: the universal protein knowledgebase. *Nucleic Acids Research*, 45(D1):D158, 2017.

[187] NCBI Resource Coordinators. Database resources of the national center for biotechnology information. *Nucleic Acids Research*, 44(D1):D7, 2016.

[188] J. A. Marsh, B. Dancheck, M. J. Ragusa, M. Allaire, J. D. Forman-Kay, and W. Peti. Structural diversity in free and bound states of intrinsically disordered protein phosphatase 1 regulators. *Structure (London, England : 1993)*, 18(9):1094–103, 2010.

[189] S. I. Walaas, D. W. Aswad, and P. Greengard. A dopamine- and cyclic AMP-regulated phosphoprotein enriched in dopamine-innervated brain regions. *Nature*, 301(5895):69–71, 1983.

[190] S. Brené, N. Lindefors, M. Ehrlich, T. Taubes, A. Horiuchi, J. Kopp, H. Hall, G. Sedvall, P. Greengard, and H. Persson. Expression of mRNAs encoding ARPP-16/19, ARPP-21, and DARPP-32 in human brain tissue. *The Journal of Neuroscience*, 14(3):985–98, 1994.

[191] M. Cerovic, R. D'Isa, R. Tonini, and R. Brambilla. Molecular and cellular mechanisms of dopamine-mediated behavioral plasticity in the striatum. *Neurobiology of Learning and*

Memory, 105:63–80, 2013.

[192] R. J. Beninger and T. V. Gerdjikov. Dopamine-Glutamate Interactions in Reward-Related Incentive Learning. In *Dopamine and Glutamate in Psychiatric Disorders*, pages 319–354. Humana Press, Totowa, NJ, 2005.

[193] M. Yger and J.-A. Girault. DARPP-32, Jack of All Trades... Master of Which? *Frontiers in Behavioral Neuroscience*, 5(56), 2011.

[194] K. Cho, M. H. Cho, J. H. Seo, J. Peak, K. H. Kong, S. Y. Yoon, and D. H. Kim. Calpain-mediated cleavage of DARPP-32 in Alzheimer's disease. *Aging Cell*, 14(5):878–886, 2015.

[195] S. D. Philibin, A. Hernandez, D. W. Self, and J. A. Bibb. Striatal signal transduction and drug addiction. *Frontiers in Neuroanatomy*, 5(60), 2011.

[196] Y. Kunii, T. M. Hyde, T. Ye, C. Li, B. Kolachana, D. Dickinson, D. R. Weinberger, J. E. Kleinman, and B. K. Lipska. Revisiting DARPP-32 in postmortem human brain: changes in schizophrenia and bipolar disorder and genetic associations with t-DARPP-32 expression. *Molecular Psychiatry*, 19:192–199, 2014.

[197] H. Wang, M. Farhan, J. Xu, P. Lazarovici, W. Zheng, H. Wang, M. Farhan, J. Xu, P. Lazarovici, W. Zheng, H. Wang, M. Farhan, J. Xu, P. Lazarovici, and W. Zheng. The involvement of DARPP-32 in the pathophysiology of schizophrenia. *Oncotarget*, 5(0), 2015.

[198] A. Nishi and T. Shuto. Potential for targeting dopamine/DARPP-32 signaling in neuropsychiatric and neurodegenerative disorders. *Expert Opinion on Therapeutic Targets*, 21 (3):259–272, 2017.

[199] A. Stipanovich, E. Valjent, M. Matamales, A. Nishi, J.-H. Ahn, M. Maroteaux, J. Bertran-Gonzalez, K. Brami-Cherrier, H. Enslen, A.-G. Corbillé, O. Filhol, A. C. Nairn, P. Greengard, D. Hervé, and J.-A. Girault. A phosphatase cascade by which rewarding stimuli control nucleosomal response. *Nature*, 453(7197):879–884, 2008.

[200] R. Kötter. Postsynaptic integration of glutamatergic and dopaminergic signals in the striatum. *Progress in Neurobiology*, 44(2):163–196, 1994.

[201] M. Lindskog, M. Kim, M. A. Wikström, K. T. Blackwell, and J. H. Kotaleski. Transient calcium and dopamine increase PKA activity and DARPP-32 phosphorylation. *PLoS computational biology*, 2(9):e119, 2006.

[202] O. Gutierrez-Arenas, O. Eriksson, and J. H. Kotaleski. Segregation and crosstalk of D1 receptor-mediated activation of ERK in striatal medium spiny neurons upon acute administration of psychostimulants. *PLoS computational biology*, 10(1):e1003445, 2014.

[203] A. G. Nair, O. Gutierrez-Arenas, O. Eriksson, A. Jauhiainen, K. T. Blackwell, and J. H. Kotaleski. Modeling intracellular signaling underlying striatal function in health and disease. *Progress in Molecular Biology and Translational Science*, 123:277–304, 2014.

[204] Z. Qi, G. W. Miller, and E. O. Voit. The internal state of medium spiny neurons varies in response to different input signals. *BMC systems biology*, 4(1):26, 2010.

[205] R. F. Oliveira, M. Kim, and K. T. Blackwell. Subcellular location of PKA controls striatal plasticity: stochastic simulations in spiny dendrites. *PLoS computational biology*, 8(2): e1002383, 2012.

[206] P. E. Barbano, M. Spivak, M. Flajolet, A. C. Nairn, P. Greengard, and L. Greengard. A

mathematical tool for exploring the dynamics of biological networks. *Proceedings of the National Academy of Sciences*, 104(49):19169–19174, 2007.

[207] J. A. Girault, H. C. J. Hemmings, K. R. Williams, A. C. Nairn, and P. Greengard. Phosphorylation of DARPP-32, a Dopamine- and cAMP-regulated Phosphoprotein, by Casein Kinase II. *Journal of Biological Chemistry*, 264(36):21748–21759, 1989.

[208] P. Svenningsson, A. Nishi, G. Fisone, J.-A. Girault, A. C. Nairn, and P. Greengard. DARPP-32: an integrator of neurotransmission. *Annual review of pharmacology and toxicology*, 44:269–96, 2004.

[209] A. Nishi, J. A. Bibb, S. Matsuyama, M. Hamada, H. Higashi, A. C. Nairn, and P. Greengard. Regulation of DARPP-32 dephosphorylation at PKA- and Cdk5-sites by NMDA and AMPA receptors: distinct roles of calcineurin and protein phosphatase-2A. *Journal of neurochemistry*, 81(4):832–41, 2002.

[210] V. Danos, H. Koeppl, and J. Wilson-Kanamori. Cooperative assembly systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6937 LNCS:1–20, 2011.

[211] M. Nic, J. Jirat, and B. Kosata. IUPAC compendium of chemical terminology (gold book), online version, 2014.

[212] S. Sivakumaran, S. Hariharaputran, J. Mishra, and U. S. Bhalla. The Database of Quantitative Cellular Signaling: management and analysis of chemical kinetic models of signaling networks. *Bioinformatics (Oxford, England)*, 19(3):408–415, 2003.

[213] S. Placzek, I. Schomburg, A. Chang, L. Jeske, M. Ulbrich, J. Tillack, and D. Schomburg. BRENDA in 2017: new perspectives and new tools in BRENDA. *Nucleic Acids Research*, 45(D1):D380–D388, 2017.

[214] U. S. Bhalla and R. Iyengar. Emergent properties of networks of biological signaling pathways. *Science*, 283(5400):381–7, 1999.

[215] S. Kuroda, N. Schweighofer, and M. Kawato. Exploration of Signal Transduction Pathways in Cerebellar Long-Term Depression by Kinetic Simulation. *Journal of Neuroscience*, 21(15), 2001.

[216] B. Dancheck, A. C. Nairn, and W. Peti. Detailed structural characterization of unbound protein phosphatase 1 inhibitors. *Biochemistry*, 47(47):12346–56, 2008.

[217] O. Engmann, A. Giralt, N. Gervasi, L. Marion-Poll, L. Gasmi, O. Filhol, M. R. Picciotto, D. Gilligan, P. Greengard, A. C. Nairn, D. Hervé, and J.-A. Girault. DARPP-32 interaction with adducin may mediate rapid environmental effects on striatal neurons. *Nature communications*, 6:10099, 2015.

[218] M. S. Choy, R. Page, and W. Peti. Regulation of protein phosphatase 1 by intrinsically disordered proteins. *Biochemical Society transactions*, 40(5):969–74, 2012.

[219] K. Takahashi, K. Kaizu, B. Hu, and M. Tomita. A multi-algorithm, multi-timescale method for cell simulation. *Bioinformatics (Oxford, England)*, 20(4):538–46, 2004.

[220] E. T. Somogyi, J.-M. Bouteiller, J. A. Glazier, M. König, J. K. Medley, M. H. Swat, and H. M. Sauro. libroadrunner: a high performance sbml simulation and analysis library. *Bioinformatics*, 31(20):3315–3321, 2015.

[221] T. R. Maarleveld, B. G. Olivier, and F. J. Bruggeman. Stochpy: A comprehensive, user-

friendly tool for simulating stochastic biological processes. *PLOS ONE*, 8(11):1–10, 2013.

[222] A. J. Lotka. Analytical note on certain rhythmic relations in organic systems. *Proceedings of the National Academy of Sciences*, 6(7):410–415, 1920.

[223] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4): 500–544, 1952.

[224] J. Lisman. A mechanism for the Hebb and the anti-Hebb processes underlying learning and memory. *Proceedings of the National Academy of Sciences of the United States of America*, 86(23):9574–8, 1989.

[225] M. K. Morris, J. Saez-Rodriguez, P. K. Sorger, and D. A. Lauffenburger. Logic-based models for the analysis of cell signaling networks. *Biochemistry*, 49(15):3216–24, 2010.

[226] C. Chaouiya. Petri net modelling of biological networks. *Briefings in bioinformatics*, 8(4): 210–9, 2007.

[227] C. A. Lo, I. Kays, F. Emran, T. J. Lin, V. Cvetkovska, and B. E. Chen. Quantification of Protein Levels in Single Living Cells. *Cell Reports*, 13(11):2634–2644, 2015.

[228] N. Otmakhov and J. Lisman. Measuring CaMKII concentration in dendritic spines. *Journal of neuroscience methods*, 203(1):106–14, 2012.

[229] F. Sacco, L. Perfetto, L. Castagnoli, and G. Cesareni. The human phosphatase interactome: An intricate family portrait. *FEBS letters*, 586(17):2732–9, 2012.

[230] N. Ozlu, B. Akten, W. Timm, N. Haseley, H. Steen, and J. A. J. Steen. Phosphoproteomics. *Wiley interdisciplinary reviews. Systems biology and medicine*, 2(3):255–76, 2010.

[231] E. Kent, S. Neumann, U. Kummer, and P. Mendes. What can we learn from global sensitivity analysis of biochemical systems? *PloS one*, 8(11):e79244, 2013.

[232] G. Lebedeva, A. Sorokin, D. Faratian, P. Mullen, A. Goltsov, S. P. Langdon, D. J. Harrison, and I. Goryanin. Model-based global sensitivity analysis as applied to identification of anti-cancer drug targets and biomarkers of drug resistance in the ErbB2/3 network. *European journal of pharmaceutical sciences : official journal of the European Federation for Pharmaceutical Sciences*, 46(4):244–58, 2012.

[233] Z. Zhang, J. C. Lee, L. Lin, V. Olivas, V. Au, T. LaFramboise, M. Abdel-Rahman, X. Wang, A. D. Levine, J. K. Rho, Y. J. Choi, C.-M. Choi, S.-W. Kim, S. J. Jang, Y. S. Park, W. S. Kim, D. H. Lee, J.-S. Lee, V. A. Miller, M. Arcila, M. Ladanyi, P. Moonsamy, C. Sawyers, T. J. Boggon, P. C. Ma, C. Costa, M. Taron, R. Rosell, B. Halmos, and T. G. Bivona. Activation of the AXL kinase causes resistance to EGFR-targeted therapy in lung cancer. *Nature Genetics*, 44(8):852–860, 2012.

[234] E. Pazarentzos and T. G. Bivona. Adaptive stress signaling in targeted cancer therapy resistance. *Oncogene*, 34(45):5599–5606, 2015.

[235] X.-Y. Zhang, M. Trame, L. Lesko, and S. Schmidt. Sobol Sensitivity Analysis: A Tool to Guide the Development and Evaluation of Systems Pharmacology Models. *CPT: Pharmacometrics & Systems Pharmacology*, 4(2):69–79, 2015.

[236] B. Iooss and A. Saltelli. Introduction to sensitivity analysis. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of Uncertainty Quantification*, pages 1–20, Cham, 2016. Springer International Publishing.

[237] C. Bishop. *Pattern Recognition and Machine Learning: All "just the Facts 101" Material*. Information science and statistics. Springer, 2013.

[238] B. Andreopoulos, A. An, X. Wang, and M. Schroeder. A roadmap of clustering algorithms: finding a match for a biomedical application. *Briefings in Bioinformatics*, 10(3): 297–314, 2008.

[239] G. V. Steeg, A. Galstyan, F. Sha, and S. DeDeo. Demystifying Information-Theoretic Clustering. *arXiv*, 32:11, 2013.

[240] C. E. Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27(April 1928):379–423, 1948.

[241] C. Chan, A. Al-bashabsheh, Q. Zhou, T. Kaced, and T. Liu. Info-Clustering : A Mathematical Theory for Data Clustering. *IEEE Transactions on Molecular, Biological, and Multi-Scale Communications; Special Issue on Biological Applications of Information Theory in Honor of Claude Shannon's Centennial*, 2(1):64–91, 2017.

[242] L. Faivishevsky and J. Goldberger. A Nonparametric Information Theoretic Clustering Algorithm. In *Proceedings of the 27th International Conference on Machine Learning*, 2010.

[243] D. Araújo, A. D. Neto, and A. Martins. Comparative study on information theoretic clustering and classical clustering algorithms. In A. E. P. Villa, W. Duch, P. Érdi, F. Masulli, and G. Palm, editors, *Artificial Neural Networks and Machine Learning – ICANN 2012*, pages 459–466, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[244] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 95(25):14863–8, 1998.

[245] G. Ver Steeg and A. Galstyan. Discovering structure in high-dimensional data through correlation explanation. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[246] C. Wiwie, J. Baumbach, and R. Röttger. Comparing the performance of biomedical clustering methods. *Nature Methods*, 12(11):1033–1038, 2015.

[247] S. Monti, P. Tamayo, J. Mesirov, and T. Golub. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning*, 52(1):91–118, 2003.

[248] S. Pepke and G. Ver Steeg. Comprehensive discovery of subsample gene expression components by information explanation: therapeutic implications in cancer. *BMC medical genomics*, 10(1):12, 2017.

[249] S. K. Madsen, G. V. Steeg, A. Mezher, N. Jahanshad, T. M. Nir, X. Hua, B. A. Gutman, A. Galstyan, and P. M. Thompson. Information-theoretic characterization of blood panel predictors for brain atrophy and cognitive decline in the elderly. *Proceedings - International Symposium on Biomedical Imaging*, 2015-July:980–984, 2015.

[250] G. Ver Steeg and A. Galstyan. Maximally informative hierarchical representations of high-dimensional data. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

[251] S. Watanabe. Information Theoretical Analysis of Multivariate Correlation. *IBM Journal of Research and Development*, 4:66–82, 1960.

[252] D. J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge Univer-

sity Press, New York, NY, USA, 2002.

[253] G. Ver Steeg. Personal communication, 2017.

[254] P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(C):53–65, 1987.

[255] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

[256] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[257] A. Saltelli. Sensitivity Analysis for Importance Assessment. *Risk Analysis*, 22(3):579–590, 2002.

[258] G. Hornberger and R. Spear. Approach to the preliminary analysis of environmental systems. *J. Environ. Manage.; (United States)*, 12:1, 1981.

[259] R. N. Gutenkunst, J. J. Waterfall, F. P. Casey, K. S. Brown, C. R. Myers, and J. P. Sethna. Universally Sloppy Parameter Sensitivities in Systems Biology Models. *PLoS Computational Biology*, 3(10):e189, 2007.

[260] B. Iooss and P. Lemaître. A review on global sensitivity analysis methods. In G. Dellino and C. Meloni, editors, *Uncertainty management in Simulation-Optimization of Complex Systems: Algorithms and Applications*, pages 101–122. Springer, 2015.

[261] Springer, European Mathematical Society. Encyclopedia of Mathematics. https://www.encyclopediaofmath.org. Accessed 2018-06-22.

[262] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global Sensitivity Analysis: The Primer*. John Wiley, 2008.

[263] L. Lilburne and S. Tarantola. Sensitivity analysis of spatial models. *International Journal of Geographical Information Science*, 23(2):151–168, 2009.

[264] J. Goffart, M. Rabouille, and N. Mendes. Uncertainty and sensitivity analysis applied to hygrothermal simulation of brick materials in a hot and humid climate. *Journal of Building Performance Simulation*, 10(1):37–57, 2015.

[265] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto. *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*. Wiley, 2004.

[266] E. Borgonovo and E. Plischke. Sensitivity analysis: A review of recent advances. *European Journal of Operational Research*, 248:869–887, 2015.

[267] Z. Zi. Sensitivity analysis approaches applied to systems biology models. *IET systems biology*, 5(6):336–6, 2011.

[268] S. Marino, I. B. Hogue, C. J. Ray, and D. E. Kirschner. A methodology for performing global uncertainty and sensitivity analysis in systems biology. *Journal of theoretical biology*, 254(1):178–96, 2008.

[269] A. Saltelli, S. Tarantola, and K. P.-S. Chan. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics*, 41(1):39–56, 1999.

[270] E. Domínguez-Hüttinger, N. J. Boon, T. B. Clarke, and R. J. Tanaka. Mathematical Modeling of Streptococcus pneumoniae Colonization, Invasive Infection and Treatment.

Frontiers in physiology, 8:115, 2017.

[271] F. Agusto, S. Bewick, and W. Fagan. Mathematical model of Zika virus with vertical transmission. *Infectious Disease Modelling*, 2(2):244–267, 2017.

[272] A. Sorokin, O. Sorokina, and J. D. Armstrong. RKappa: Statistical Sampling Suite for Kappa Models. In O. Maler, H. Ádám, T. Dang, and C. Piazza, editors, *Hybrid Systems Biology. Lecture Notes in Computer Science*, volume 7699, pages 128–142. Springer, Cham, 2015.

[273] A. Sorokin. R4Kappa: library for statistical sampling of Kappa parameter space. https://github.com/lptolik/R4Kappa, 2016. Accessed 2016-11-01.

[274] G. Pujol, B. Iooss, A. J. with contributions from Khalid Boumhaout, S. D. Veiga, T. Delage, J. Fruth, L. Gilquin, J. Guillaume, L. Le Gratiet, P. Lemaitre, B. L. Nelson, F. Monari, R. Oomen, B. Ramos, O. Roustant, E. Song, J. Staum, T. Touati, and F. Weber. *sensitivity: Global Sensitivity Analysis of Model Outputs*, 2017. R package version 1.15.0.

[275] J. P. C. Kleijnen and J. C. Helton. Statistical analyses of scatterplots to identify important factors in large-scale simulations, 1: Review and comparison of techniques. *Reliability Engineering & System Safety*, 65(2):147–185, 1999.

[276] R. I. Cukier, C. M. Fortuin, K. E. Shuler, A. G. Petschek, and J. H. Schaibly. Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. I Theory. *The Journal of Chemical Physics*, 59(8):3873–3878, 1973.

[277] I. M. Sobol and S. S. Kucherenko. Global sensitivity indices for nonlinear mathematical models. Review. *Wilmott*, 2005(1):56–61, 2005.

[278] T. Homma and A. Saitelip. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering and System Sa]ety*, 52(1), 1996.

[279] E. Borgonovo. A new uncertainty importance measure. *Reliability Engineering and System Safety*, 2007.

[280] M. F. Moody. Chapter 3 - fourier fundamentals. In M. F. Moody, editor, *Structural Biology Using Electrons and X-rays*, pages 25 – 54. Academic Press, Boston, 2011.

[281] J.-Y. Tissot and C. Prieur. Bias correction for the estimation of sensitivity indices based on random balance designs. *Reliability Engineering & System Safety*, 107:205–213, 2012.

[282] S. Da Veiga. Global sensitivity analysis with dependence measures. *Journal of Statistical Computation and Simulation*, 85(7):1283–1305, 2015.

[283] M. Baucells and E. Borgonovo. Invariant Probabilistic Sensitivity Analysis. *Management Science*, 59(11):2536–2549, 2013.

[284] S. Sinha. Hilbert-Schmidt and Sobol sensitivity indices for static and time series Wnt signaling measurements in colorectal cancer - part A. *BMC Systems Biology*, 11(1):120, 2017.

[285] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In S. Jain, H. U. Simon, and E. Tomita, editors, *Algorithmic Learning Theory*, pages 63–77, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[286] M. De Lozzo and A. Marrel. New improvements in the use of dependence measures for sensitivity analysis and screening. *Journal of Statistical Computation and Simulation*, 86 (15):3038–3058, 2016.

[287] A. Smola, A. Gretton, L. Song, and B. Schölkopf. A Hilbert Space Embedding for Distributions. In *Discovery Science*, pages 40–41. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[288] M. De Lozzo and A. Marrel. Sensitivity analysis with dependence and variance-based measures for spatio-temporal numerical simulators. *Stochastic Environmental Research and Risk Assessment*, 31(6):1437–1453, 2017.

[289] E. Atanassov, A. Karaivanova, and S. Ivanovska. Tuning the generation of sobol sequence with owen scrambling. In I. Lirkov, S. Margenov, and J. Waśniewski, editors, *Large-Scale Scientific Computing*, pages 459–466, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[290] C. Dutang and P. Savicky. Quick introduction of randtoolbox, 2009.

[291] Open grid scheduler. http://gridscheduler.sourceforge.net. Accessed 2017-12-30.

[292] R. L. Iman and W. J. Conover. A measure of top – down correlation. *Technometrics*, 29(3): 351–357, 1987.

[293] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python. http://www.scipy.org, 2001. Accessed 2017-12-17.

[294] T. Ideker and N. J. Krogan. Differential network biology. *Molecular systems biology*, 8:565, 2012.

[295] Adjustment for chance in clustering performance evaluation. http://scikit-learn.org/stable/auto_examples/cluster/plot_adjusted_for_chance_measures.html. Accessed 2017-12-10.

[296] J. H. Ward. Hierarchical Grouping to Optimize an Objective Function, 1963.

[297] Y. Lichtblau, K. Zimmermann, B. Haldemann, D. Lenze, M. Hummel, and U. Leser. Comparative assessment of differential network analysis methods. *Briefings in Bioinformatics*, page bbw061, 2016.

[298] T. F. Fuller, A. Ghazalpour, J. E. Aten, T. A. Drake, A. J. Lusis, and S. Horvath. Weighted gene coexpression network analysis strategies applied to mouse weight. *Mammalian genome : official journal of the International Mammalian Genome Society*, 18(6-7):463–72, 2007.

[299] D. Ruan, A. Young, and G. Montana. Differential analysis of biological networks. *BMC Bioinformatics*, 16(1):327, 2015.

[300] R. Mall, L. Cerulo, H. Bensmail, A. Iavarone, and M. Ceccarelli. Detection of statistically significant network changes in complex biological networks. *BMC systems biology*, 11(1): 32, 2017.

[301] S. Prabakaran, G. Lippens, H. Steen, and J. Gunawardena. Post-translational modification: Nature's escape from genetic imprisonment and the basis for dynamic information encoding. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 4(6):565–583, 2012.

[302] J. Nealon, L. Philomina, and L. McGuffin. Predictive and Experimental Approaches for Elucidating Protein–Protein Interactions and Quaternary Structures. *International Journal of Molecular Sciences*, 18(12):2623, 2017.

[303] J. R. Parrish, K. D. Gulyas, and R. L. Finley. Yeast two-hybrid contributions to interactome mapping. *Current opinion in biotechnology*, 17(4):387–93, 2006.

[304] H. N. Chua and L. Wong. Increasing the reliability of protein interactomes. *Drug Discovery Today*, 13(15-16):652–658, 2008.

[305] E. Sprinzak, S. Sattath, and H. Margalit. How Reliable are Experimental Protein–Protein Interaction Data? *Journal of Molecular Biology*, 327(5):919–923, 2003.

[306] M. Singhal and H. Resat. A domain-based approach to predict protein-protein interactions. *BMC bioinformatics*, 8:199, 2007.

[307] D. Petrey and B. Honig. Structural bioinformatics of the interactome. *Annual review of biophysics*, 43:193–210, 2014.

[308] G. Alanis-Lobato, M. A. Andrade-Navarro, and M. H. Schaefer. HIPPIE v2.0: enhancing meaningfulness and reliability of protein–protein interaction networks. *Nucleic Acids Research*, 45(D1):D408–D414, 2017.

[309] H. Hermjakob, L. Montecchi-Palazzi, G. Bader, J. Wojcik, L. Salwinski, A. Ceol, S. Moore, S. Orchard, U. Sarkans, C. von Mering, B. Roechert, S. Poux, E. Jung, H. Mersch, P. Kersey, M. Lappe, Y. Li, R. Zeng, D. Rana, M. Nikolski, H. Husi, C. Brun, K. Shanker, S. G. N. Grant, C. Sander, P. Bork, W. Zhu, A. Pandey, A. Brazma, B. Jacq, M. Vidal, D. Sherman, P. Legrain, G. Cesareni, I. Xenarios, D. Eisenberg, B. Steipe, C. Hogue, and R. Apweiler. The HUPO PSI's molecular interaction format–a community standard for the representation of protein interaction data. *Nature biotechnology*, 22(2):177–83, 2004.

[310] S. Kerrien, S. Orchard, L. Montecchi-Palazzi, B. Aranda, A. F. Quinn, N. Vinod, G. D. Bader, I. Xenarios, J. Wojcik, D. Sherman, M. Tyers, J. J. Salama, S. Moore, A. Ceol, A. Chatr-aryamontri, M. Oesterheld, V. Stümpflen, L. Salwinski, J. Nerothin, E. Cerami, M. E. Cusick, M. Vidal, M. Gilson, J. Armstrong, P. Woollard, C. Hogue, D. Eisenberg, G. Cesareni, R. Apweiler, and H. Hermjakob. Broadening the horizon – level 2.5 of the HUPO-PSI format for molecular interactions. *BMC Biology*, 5(1):44, 2007.

[311] B. Aranda, H. Blankenburg, S. Kerrien, F. S. L. Brinkman, A. Ceol, E. Chautard, J. M. Dana, J. De Las Rivas, M. Dumousseau, E. Galeota, A. Gaulton, J. Goll, R. E. W. Hancock, R. Isserlin, R. C. Jimenez, J. Kerssemakers, J. Khadake, D. J. Lynn, M. Michaut, G. O'Kelly, K. Ono, S. Orchard, C. Prieto, S. Razick, O. Rigina, L. Salwinski, M. Simonovic, S. Velankar, A. Winter, G. Wu, G. D. Bader, G. Cesareni, I. M. Donaldson, D. Eisenberg, G. J. Kleywegt, J. Overington, S. Ricard-Blum, M. Tyers, M. Albrecht, and H. Hermjakob. PSICQUIC and PSISCORE: accessing and scoring molecular interactions. *Nature methods*, 8(7):528–9, 2011.

[312] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic acids research*, 39(Database issue):D52–7, 2011.

[313] V. Memišević, A. Wallqvist, and J. Reifman. Reconstituting protein interaction networks using parameter-dependent domain-domain interactions. *BMC bioinformatics*, 14:154, 2013.

[314] R. P. Sheridan and R. Venkataraghavan. A systematic search for protein signature sequences. *Proteins*, 14(1):16–28, 1992.

[315] R. Apweiler, T. K. Attwood, A. Bairoch, A. Bateman, E. Birney, M. Biswas, P. Bucher, L. Cerutti, F. Corpet, M. D. Croning, R. Durbin, L. Falquet, W. Fleischmann, J. Gouzy, H. Hermjakob, N. Hulo, I. Jonassen, D. Kahn, A. Kanapin, Y. Karavidopoulou, R. Lopez,

B. Marx, N. J. Mulder, T. M. Oinn, M. Pagni, F. Servant, C. J. Sigrist, and E. M. Zdobnov. The InterPro database, an integrated documentation resource for protein families, domains and functional sites. *Nucleic acids research*, 29(1):37–40, 2001.

[316] A. Mitchell, H.-Y. Chang, L. Daugherty, M. Fraser, S. Hunter, R. Lopez, C. McAnulla, C. McMenamin, G. Nuka, S. Pesseat, A. Sangrador-Vegas, M. Scheremetjew, C. Rato, S.-Y. Yong, A. Bateman, M. Punta, T. K. Attwood, C. J. A. Sigrist, N. Redaschi, C. Rivoire, I. Xenarios, D. Kahn, D. Guyot, P. Bork, I. Letunic, J. Gough, M. Oates, D. Haft, H. Huang, D. A. Natale, C. H. Wu, C. Orengo, I. Sillitoe, H. Mi, P. D. Thomas, and R. D. Finn. The InterPro protein families database: the classification resource after 15 years. *Nucleic acids research*, 43(Database issue):D213–21, 2015.

[317] S. R. Eddy. Profile hidden Markov models. *Bioinformatics*, 14(9):755–763, 1998.

[318] R. D. Finn, T. K. Attwood, P. C. Babbitt, A. Bateman, P. Bork, A. J. Bridge, H.-Y. Chang, Z. Dosztányi, S. El-Gebali, M. Fraser, J. Gough, D. Haft, G. L. Holliday, H. Huang, X. Huang, I. Letunic, R. Lopez, S. Lu, A. Marchler-Bauer, H. Mi, J. Mistry, D. A. Natale, M. Necci, G. Nuka, C. A. Orengo, Y. Park, S. Pesseat, D. Piovesan, S. C. Potter, N. D. Rawlings, N. Redaschi, L. Richardson, C. Rivoire, A. Sangrador-Vegas, C. Sigrist, I. Sillitoe, B. Smithers, S. Squizzato, G. Sutton, N. Thanki, P. D. Thomas, S. C. E. Tosatto, C. H. Wu, I. Xenarios, L.-S. Yeh, S.-Y. Young, and A. L. Mitchell. InterPro in 2017-beyond protein family and domain annotations. *Nucleic acids research*, 45(D1):D190–D199, 2017.

[319] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Research*, 28(1):235–242, 2000.

[320] Y. Kim, B. Min, and G.-S. Yi. IDDI: integrated domain-domain interaction and protein interaction analysis system. *Proteome science*, 10 Suppl 1(Suppl 1):S9, 2012.

[321] R. D. Finn, B. L. Miller, J. Clements, and A. Bateman. iPfam: a database of protein family and domain interactions found in the Protein Data Bank. *Nucleic Acids Research*, 42(Database issue):D364, 2014.

[322] A. Stein, R. B. Russell, and P. Aloy. 3did: interacting protein domains of known three-dimensional structure. *Nucleic acids research*, 33(Database issue):D413–7, 2005.

[323] R. Mosca, A. Céol, A. Stein, R. Olivella, and P. Aloy. 3did: a catalog of domain-based interactions of known three-dimensional structure. *Nucleic acids research*, 42(Database issue):D374–9, 2014.

[324] A. J. Bordner and A. A. Gorin. Comprehensive inventory of protein complexes in the Protein Data Bank from consistent classification of interfaces. *BMC Bioinformatics*, 9:234, 2008.

[325] S. Khor. Inferring domain-domain interactions from protein-protein interactions with formal concept analysis. *PloS one*, 9(2):e88943, 2014.

[326] S. Yellaboina, A. Tasneem, D. V. Zaykin, B. Raghavachari, and R. Jothi. DOMINE: a comprehensive collection of known and predicted domain-domain interactions. *Nucleic acids research*, 39(Database issue):D730–5, 2011.

[327] P. Björkholm and E. L. L. Sonnhammer. Comparative analysis and unification of domain–domain interaction networks. *Bioinformatics*, 25(22):3020–3025, 2009.

[328] Q. Luo, P. Pagel, B. Vilne, and D. Frishman. DIMA 3.0: Domain Interaction Map. *Nucleic*

acids research, 39(Database issue):D724–9, 2011.

[329] K. Oh and G.-S. Yi. Prediction of scaffold proteins based on protein interaction and domain architectures. *BMC Bioinformatics*, 17(6):220, 2016.

[330] L.-L. Zheng, C. Li, J. Ping, Y. Zhou, Y. Li, and P. Hao. The domain landscape of virus-host interactomes. *BioMed research international*, 2014:867235, 2014.

[331] B. Kholodenko, M. B. Yaffe, and W. Kolch. Computational approaches for analyzing information flow in biological networks. *Science signaling*, 5(220):re1, 2012.

[332] S. Lemeer and A. J. R. Heck. The phosphoproteomics data explosion. *Current opinion in chemical biology*, 13(4):414–20, 2009.

[333] A. Bensimon, A. J. R. Heck, and R. Aebersold. Mass spectrometry-based proteomics and network biology. *Annual review of biochemistry*, 81:379–405, 2012.

[334] C. Choudhary and M. Mann. Decoding signalling networks by mass spectrometry-based proteomics. *Nature reviews. Molecular cell biology*, 11(6):427–39, 2010.

[335] B. Trost and A. Kusalik. Computational prediction of eukaryotic phosphorylation sites. *Bioinformatics (Oxford, England)*, 27(21):2927–35, 2011.

[336] H. Horn, E. M. Schoof, J. Kim, X. Robin, M. L. Miller, F. Diella, A. Palma, G. Cesareni, L. J. Jensen, and R. Linding. KinomeXplorer: an integrated platform for kinome biology studies. *Nature Methods*, 11(6):603–604, 2014.

[337] R. H. Newman, J. Hu, H.-S. Rho, Z. Xie, C. Woodard, J. Neiswinger, C. Cooper, M. Shirley, H. M. Clark, S. Hu, W. Hwang, J. S. Jeong, G. Wu, J. Lin, X. Gao, Q. Ni, R. Goel, S. Xia, H. Ji, K. N. Dalby, M. J. Birnbaum, P. A. Cole, S. Knapp, A. G. Ryazanov, D. J. Zack, S. Blackshaw, T. Pawson, A.-C. Gingras, S. Desiderio, A. Pandey, B. E. Turk, J. Zhang, H. Zhu, and J. Qian. Construction of human activity-based phosphorylation networks. *Molecular systems biology*, 9:655, 2013.

[338] J. Hu, H.-S. Rho, R. H. Newman, J. Zhang, H. Zhu, and J. Qian. PhosphoNetworks: a database for human phosphorylation networks. *Bioinformatics (Oxford, England)*, 30(1): 141–2, 2014.

[339] P. V. Hornbeck, J. M. Kornhauser, S. Tkachev, B. Zhang, E. Skrzypek, B. Murray, V. Latham, and M. Sullivan. PhosphoSitePlus: a comprehensive resource for investigating the structure and function of experimentally determined post-translational modifications in man and mouse. *Nucleic Acids Research*, 40(D1):D261–D270, 2012.

[340] F. Gnad, J. Gunawardena, and M. Mann. PHOSIDA 2011: the posttranslational modification database. *Nucleic Acids Research*, 39(Database):D253–D260, 2011.

[341] F. Diella, S. Cameron, C. Gemünd, R. Linding, A. Via, B. Kuster, T. Sicheritz-Pontén, N. Blom, and T. J. Gibson. Phospho.ELM: A database of experimentally verified phosphorylation sites in eukaryotic proteins. *BMC Bioinformatics*, 5(1):79, 2004.

[342] F. Büchel, N. Rodriguez, N. Swainston, C. Wrzodek, T. Czauderna, R. Keller, F. Mittag, M. Schubert, M. Glont, M. Golebiewski, M. van Iersel, S. Keating, M. Rall, M. Wybrow, H. Hermjakob, M. Hucka, D. B. Kell, W. Müller, P. Mendes, A. Zell, C. Chaouiya, J. Saez-Rodriguez, F. Schreiber, C. Laibe, A. Dräger, and N. Le Novère. Path2Models: large-scale generation of computational models from biochemical pathway maps. *BMC systems biology*, 7(1):116, 2013.

[343] C. Wrzodek, F. Büchel, M. Ruff, A. Dräger, and A. Zell. Precise generation of systems biology models from KEGG pathways. *BMC systems biology*, 7(1):15, 2013.

[344] R. Caspi, T. Altman, R. Billington, K. Dreher, H. Foerster, C. A. Fulcher, T. A. Holland, I. M. Keseler, A. Kothari, A. Kubo, M. Krummenacker, M. Latendresse, L. A. Mueller, Q. Ong, S. Paley, P. Subhraveti, D. S. Weaver, D. Weerasinghe, P. Zhang, and P. D. Karp. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases. *Nucleic Acids Research*, 42(D1):D459–D471, 2014.

[345] U. Wittig, R. Kania, M. Golebiewski, M. Rey, L. Shi, L. Jong, E. Algaa, A. Weidemann, H. Sauer-Danzwith, S. Mir, O. Krebs, M. Bittkowski, E. Wetsch, I. Rojas, and W. Müller. SABIO-RK–database for biochemical reaction kinetics. *Nucleic acids research*, 40(Database issue):D790–6, 2012.

[346] A. Fabregat, K. Sidiropoulos, P. Garapati, M. Gillespie, K. Hausmann, R. Haw, B. Jassal, S. Jupe, F. Korninger, S. McKay, L. Matthews, B. May, M. Milacic, K. Rothfels, V. Shamovsky, M. Webber, J. Weiser, M. Williams, G. Wu, L. Stein, H. Hermjakob, and P. D'Eustachio. The reactome pathway knowledgebase. *Nucleic Acids Research*, 44(D1): D481, 2016.

[347] M. L. Hines, T. Morse, M. Migliore, N. T. Carnevale, and G. M. Shepherd. ModelDB: A Database to Support Computational Neuroscience. *Journal of computational neuroscience*, 17(1):7–11, 2004.

[348] D. A. Beard, R. Britten, M. T. Cooling, A. Garny, M. D. Halstead, P. J. Hunter, J. Lawson, C. M. Lloyd, J. Marsh, A. Miller, D. P. Nickerson, P. M. Nielsen, T. Nomura, S. Subramanium, S. M. Wimalaratne, and T. Yu. CellML metadata standards, associated tools and repositories. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1895):1845–1867, 2009.

[349] C. Li, M. Courtot, N. Le Novère, and C. Laibe. BioModels.net Web Services, a free and integrated toolkit for computational modelling software. *Briefings in bioinformatics*, 11 (3):270–7, 2010.

[350] S. Jupp, J. Malone, J. Bolleman, M. Brandizi, M. Davies, L. Garcia, A. Gaulton, S. Gehant, C. Laibe, N. Redaschi, S. M. Wimalaratne, M. Martin, N. Le Novère, H. Parkinson, E. Birney, and A. M. Jenkinson. The EBI RDF platform: linked open data for the life sciences. *Bioinformatics (Oxford, England)*, 30(9):1338–9, 2014.

[351] S. M. Wimalaratne, P. Grenon, H. Hermjakob, N. Le Novère, and C. Laibe. BioModels linked dataset. *BMC Systems Biology*, 8(1):91, 2014.

[352] B. Franke, S. V. Faraone, P. Asherson, J. Buitelaar, C. H. D. Bau, J. A. Ramos-Quiroga, E. Mick, E. H. Grevet, S. Johansson, J. Haavik, K.-P. Lesch, B. Cormand, and A. Reif. The genetics of attention deficit/hyperactivity disorder in adults, a review. *Molecular psychiatry*, 17(10):960–87, 2012.

[353] S. V. Faraone and E. Mick. Molecular genetics of attention deficit hyperactivity disorder. *The Psychiatric clinics of North America*, 33(1):159–80, 2010.

[354] D. Wallis, H. F. Russell, and M. Muenke. Review: Genetics of attention deficit/hyperactivity disorder. *Journal of Pediatric Psychology*, 33(10):1085–1099, 2008.

[355] H.-C. Steinhausen. The heterogeneity of causes and courses of attention-

deficit/hyperactivity disorder. *Acta Psychiatrica Scandinavica*, 120(5):392–399, 2009.

[356] A. Bari and T. W. Robbins. Animal Models of ADHD. In J. J. Hagan, editor, *Molecular and Functional Models in Neuropsychiatry*, pages 149–185. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[357] P. Majdak, J. R. Ossyra, J. M. Ossyra, A. J. Cobert, G. C. Hofmann, S. Tse, B. Panozzo, E. L. Grogan, A. Sorokina, and J. S. Rhodes. A new mouse model of ADHD for medication development. *Scientific Reports*, 6(1):39472, 2016.

[358] S. V. Faraone and H. Larsson. Genetics of attention deficit hyperactivity disorder, 2018.

[359] Q. Gao, L. Liu, Q. Qian, and Y. Wang. Advances in molecular genetic studies of attention deficit hyperactivity disorder in China. *Shanghai archives of psychiatry*, 26(4):194–206, 2014.

[360] N. Rappaport, N. Nativ, G. Stelzer, M. Twik, Y. Guan-Golan, T. I. Stein, I. Bahir, F. Belinky, C. P. Morrey, M. Safran, and D. Lancet. MalaCards: an integrated compendium for diseases and their annotation. *Database : the journal of biological databases and curation*, 2013:bat018, 2013.

[361] M. J. Landrum, J. M. Lee, G. R. Riley, W. Jang, W. S. Rubinstein, D. M. Church, and D. R. Maglott. ClinVar: public archive of relationships among sequence variation and human phenotype. *Nucleic acids research*, 42(Database issue):D980–5, 2014.

[362] A. Hamosh, A. F. Scott, J. S. Amberger, C. A. Bocchini, and V. A. McKusick. Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic acids research*, 33(Database issue):D514–7, 2005.

[363] S. Pletscher-Frankild, A. Pallejà, K. Tsafou, J. X. Binder, and L. J. Jensen. DISEASES: Text mining and data integration of disease-gene associations. *Methods*, 74:83–89, 2015.

[364] X. He and T. I. Simpson. statbio/OntoSuite-Miner: OntoSuite-Miner (Version v1.0). https://doi.org/10.5281/zenodo.819726, 2017. Accessed 2017-06-20.

[365] J. A. Mitchell, A. R. Aronson, J. G. Mork, L. C. Folk, S. M. Humphrey, and J. M. Ward. Gene indexing: characterization and analysis of NLM's GeneRIFs. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2003:460–4, 2003.

[366] A. R. Aronson and F.-M. Lang. An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236, 2010.

[367] P. L. Whetzel, N. F. Noy, N. H. Shah, P. R. Alexander, C. Nyulas, T. Tudorache, and M. A. Musen. BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic acids research*, 39(Web Server issue):W541–5, 2011.

[368] L. M. Schriml, C. Arze, S. Nadendla, Y.-W. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe. Disease Ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(Database issue):D940–6, 2012.

[369] W. A. Kibbe, C. Arze, V. Felix, E. Mitraka, E. Bolton, G. Fu, C. J. Mungall, J. X. Binder, J. Malone, D. Vasant, H. Parkinson, and L. M. Schriml. Disease Ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data. *Nucleic acids research*, 43(Database issue):D1071–8, 2015.

[370] S. Kerrien, B. Aranda, L. Breuza, A. Bridge, F. Broackes-Carter, C. Chen, M. Duesbury, M. Dumousseau, M. Feuermann, U. Hinz, C. Jandrasits, R. C. Jimenez, J. Khadake, U. Mahadevan, P. Masson, I. Pedruzzi, E. Pfeiffenberger, P. Porras, A. Raghunath, B. Roechert, S. Orchard, and H. Hermjakob. The IntAct molecular interaction database in 2012. *Nucleic acids research*, 40(Database issue):D841–6, 2012.

[371] A. Chatr-Aryamontri, B. J. Breitkreutz, R. Oughtred, L. Boucher, S. Heinicke, D. Chen, C. Stark, A. Breitkreutz, N. Kolas, L. O'Donnell, T. Reguly, J. Nixon, L. Ramage, A. Winter, A. Sellam, C. Chang, J. Hirschman, C. Theesfeld, J. Rust, M. S. Livstone, K. Dolinski, and M. Tyers. The BioGRID interaction database: 2015 update. *Nucleic Acids Research*, 43(D1):D470–D478, 2015.

[372] I. Xenarios, D. W. Rice, L. Salwinski, M. K. Baron, E. M. Marcotte, and D. Eisenberg. DIP: the database of interacting proteins. *Nucleic acids research*, 28(1):289–91, 2000.

[373] R. D. Finn, P. Coggill, R. Y. Eberhardt, S. R. Eddy, J. Mistry, A. L. Mitchell, S. C. Potter, M. Punta, M. Qureshi, A. Sangrador-Vegas, G. A. Salazar, J. Tate, and A. Bateman. The Pfam protein families database: towards a more sustainable future. *Nucleic Acids Research*, 44(D1):D279–D285, 2016.

[374] S. Fortunato and M. Barthelemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences of the United States of America*, 104(1):36–41, 2007.

[375] A. Alexa, J. Rahnenführer, and T. Lengauer. Improved scoring of functional groups from gene expression data by decorrelating GO graph structure. *Bioinformatics*, 22(13): 1600–1607, 2006.

[376] X. He and T. I. Simpson. statbio/topOnto: topOnto (Version v1.0). https://doi.org/10.5281/zenodo.819735, 2017. Accessed 2017-06-20.

[377] S. Grossmann, S. Bauer, P. N. Robinson, and M. Vingron. Improved detection of overrepresentation of Gene-Ontology annotations with parent child analysis. *Bioinformatics (Oxford, England)*, 23(22):3024–31, 2007.

[378] Y. Benjamini and D. Yekutieli. The control of the false discovery rate in multiple testing under depencency. *The Annals of Statistics*, 29(4):1165–1188, 2001.

[379] J. S. Mattick and I. V. Makunin. Non-coding RNA. *Human Molecular Genetics*, 15(suppl_1): R17–R29, 2006.

[380] Y. Tutar. Pseudogenes. *Comparative and functional genomics*, 2012:424526, 2012.

[381] M. B. Sass, A. N. Lorenz, R. L. Green, and R. A. Coleman. A pragmatic approach to biochemical systems theory applied to an alpha-synuclein-based model of parkinson's disease. *Journal of neuroscience methods*, 178(2):366—377, 2009.

[382] J. Snider, M. Kotlyar, P. Saraon, Z. Yao, I. Jurisica, and I. Stagljar. Fundamentals of protein interaction network mapping. *Molecular systems biology*, 11(12):848, 2015.

[383] M. W. Gonzalez and M. G. Kann. Chapter 4: Protein Interactions and Disease. *PLoS Computational Biology*, 8(12):e1002819, 2012.

[384] M. Oti, B. Snel, M. A. Huynen, and H. G. Brunner. Predicting disease genes using protein-protein interactions. *Journal of medical genetics*, 43(8):691–8, 2006.

[385] J. Das, R. Fragoza, H. R. Lee, N. A. Cordero, Y. Guo, M. J. Meyer, T. V. Vo, X. Wang, and H. Yu. Exploring mechanisms of human disease through structurally resolved protein

interactome networks. *Molecular bioSystems*, 10(1):9–17, 2014.

[386] M. Gustafsson, C. E. Nestor, H. Zhang, A.-L. Barabási, S. Baranzini, S. Brunak, K. F. Chung, H. J. Federoff, A.-C. Gavin, R. R. Meehan, P. Picotti, M. À. Pujana, N. Rajewsky, K. G. Smith, P. J. Sterk, P. Villoslada, and M. Benson. Modules, networks and systems medicine for understanding disease and aiding diagnosis. *Genome medicine*, 6(10):82, 2014.

[387] K. Lage. Protein–protein interactions and genetic diseases: The interactome. *Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease*, 1842(10):1971–1980, 2014.

[388] Y. Wang, Z. Liu, H. Cheng, T. Gao, Z. Pan, Q. Yang, A. Guo, and Y. Xue. EKPD: A hierarchical database of eukaryotic protein kinases and protein phosphatases. *Nucleic Acids Research*, 42(D1):D496–D502, 2014.

[389] K. Blum, A. L.-C. Chen, E. R. Braverman, D. E. Comings, T. J. H. Chen, V. Arcuri, S. H. Blum, B. W. Downs, R. L. Waite, A. Notaro, J. Lubar, L. Williams, T. J. Prihoda, T. Palomo, and M. Oscar-Berman. Attention-deficit-hyperactivity disorder and reward deficiency syndrome. *Neuropsychiatric disease and treatment*, 4(5):893–918, 2008.

[390] N. del Campo, U. Müller, and B. J. Sahakian. Neural and Behavioral Endophenotypes in ADHD. *Current Topics in Behavioural Neuroscience*, 11(April):65–91, 2012.

[391] S. Nagamitsu, Y. Yamashita, H. Tanigawa, H. Chiba, H. Kaida, M. Ishibashi, T. Kakuma, P. E. Croarkin, and T. Matsuishi. Upregulated GABA inhibitory function in ADHD children with child behavior checklist-dysregulation profile: 123i-iomazenil SPECT study. *Frontiers in Psychiatry*, 6:84, 2015.

[392] R. A. E. Edden, D. Crocetti, H. Zhu, D. L. Gilbert, and S. H. Mostofsky. Reduced GABA concentration in attention-deficit/hyperactivity disorder. *Archives of General Psychiatry*, 69(7):750–753, 2012.

[393] Z. Gu and J. Wang. CePa: an R package for finding significant pathways weighted by multiple network centralities. *Bioinformatics (Oxford, England)*, 29(5):658–60, 2013.

[394] C. Mitrea, Z. Taghavi, B. Bokanizad, S. Hanoudi, R. Tagett, M. Donato, C. Voichiţa, and S. Drăghici. Methods and approaches in the topology-based analysis of biological pathways. *Frontiers in physiology*, 4:278, 2013.

[395] I. Schomburg, A. Chang, S. Placzek, C. Söhngen, M. Rother, M. Lang, C. Munaretto, S. Ulas, M. Stelzer, A. Grote, M. Scheer, and D. Schomburg. BRENDA in 2013: integrated reactions, kinetic data, enzyme function data, improved disease classification: new options and contents in BRENDA. *Nucleic acids research*, 41(Database issue):D764–72, 2013.

[396] A. Fleischmann, M. Darsow, K. Degtyarenko, W. Fleischmann, S. Boyce, K. B. Axelsen, A. Bairoch, D. Schomburg, K. F. Tipton, and R. Apweiler. IntEnz, the integrated relational enzyme database. *Nucleic acids research*, 32(Database issue):D434–7, 2004.

[397] R. Caspi, T. Altman, K. Dreher, C. A. Fulcher, P. Subhraveti, I. M. Keseler, A. Kothari, M. Krummenacker, M. Latendresse, L. A. Mueller, Q. Ong, S. Paley, A. Pujar, A. G. Shearer, M. Travers, D. Weerasinghe, P. Zhang, and P. D. Karp. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic acids research*, 40(Database issue):D742–53, 2012.

[398] J. Hakenberg, S. Schmeier, A. Kowald, E. Klipp, and U. Leser. Finding kinetic parameters using text mining. *Omics : a journal of integrative biology*, 8(2):131–52, 2004.

[399] R. S. Costa, A. Veríssimo, and S. Vinga. KiMoSys: a web-based repository of experimental data for KInetic MOdels of biological SYStems. *BMC systems biology*, 8:85, 2014.

[400] E. A. R. Serin, H. Nijveen, H. W. M. Hilhorst, and W. Ligterink. Learning from Co-expression Networks: Possibilities and Challenges. *Frontiers in Plant Science*, 7:444, 2016.

[401] Y. Tian, B. Stamova, B. P. Ander, G. C. Jickling, J. R. Gunther, B. A. Corbett, N. G. Bos-Veneman, P. J. Hoekstra, J. B. Schweitzer, and F. R. Sharp. Correlations of gene expression with ratings of inattention and hyperactivity/impulsivity in tourette syndrome: A pilot study. *BMC Medical Genomics*, 5:49, 2012.

[402] M. Hofmann-Apitius, G. Ball, S. Gebel, S. Bagewadi, B. De Bono, R. Schneider, M. Page, A. T. Kodamullil, E. Younesi, C. Ebeling, J. Tegnér, and L. Canard. Bioinformatics mining and modeling methods for the identification of disease mechanisms in neurodegenerative disorders. *International Journal of Molecular Sciences*, 16(12):29179–29206, 2015.

[403] R. Mani, R. P. St.Onge, J. L. Hartman, G. Giaever, and F. P. Roth. Defining genetic interaction. *Proceedings of the National Academy of Sciences*, 105(9):3461–3466, 2008.

[404] M. T. Weirauch. Gene coexpression networks for the analysis of dna microarray data. In *Applied Statistics for Network Biology*, chapter 11, pages 215–250. Wiley-Blackwell, 2011.

[405] J. Y. Young, J. D. Westbrook, Z. Feng, E. Peisach, I. Persikova, R. Sala, S. Sen, J. M. Berrisford, G. J. Swaminathan, T. J. Oldfield, A. Gutmanas, R. Igarashi, D. R. Armstrong, K. Baskaran, L. Chen, M. Chen, A. R. Clark, L. Di Costanzo, D. Dimitropoulos, G. Gao, S. Ghosh, S. Gore, V. Guranovic, P. M. S. Hendrickx, B. P. Hudson, Y. Ikegawa, Y. Kengaku, C. L. Lawson, Y. Liang, L. Mak, A. Mukhopadhyay, B. Narayanan, K. Nishiyama, A. Patwardhan, G. Sahni, E. Sanz-García, J. Sato, M. R. Sekharan, C. Shao, O. S. Smart, L. Tan, G. van Ginkel, H. Yang, M. A. Zhuravleva, J. L. Markley, H. Nakamura, G. Kurisu, G. J. Kleywegt, S. Velankar, H. M. Berman, and S. K. Burley. Worldwide Protein Data Bank biocuration supporting open access to high-quality 3D structural biology data. *Database*, 2018, 2018.

[406] A. Goncearenco, B. A. Shoemaker, D. Zhang, A. Sarychev, and A. R. Panchenko. Coverage of protein domain families with structural protein-protein interactions: current progress and future trends. *Progress in biophysics and molecular biology*, 116(2-3):187–93, 2014.

[407] B. A. Shoemaker, D. Zhang, M. Tyagi, R. R. Thangudu, J. H. Fong, A. Marchler-Bauer, S. H. Bryant, T. Madej, and A. R. Panchenko. IBIS (Inferred Biomolecular Interaction Server) reports, predicts and integrates multiple types of conserved interactions for proteins. *Nucleic Acids Research*, 40(D1):D834–D840, 2012.

[408] X. Wang, X. Wei, B. Thijssen, J. Das, S. M. Lipkin, and H. Yu. Three-dimensional reconstruction of protein networks provides insight into human genetic disease. *Nat.Biotechnol.*, 30(1546-1696 (Electronic)):159–164, 2012.

[409] M. J. Meyer, J. Das, X. Wang, and H. Yu. INstruct: A database of high-quality 3D structurally resolved protein interactome networks. *Bioinformatics*, 29(12):1577–1579, 2013.

[410] S. Ullah, S. Lin, Y. Xu, W. Deng, L. Ma, Y. Zhang, Z. Liu, and Y. Xue. DbPAF: An

integrative database of protein phosphorylation in animals and fungi. *Scientific Reports*, 6(March):1–9, 2016.

[411] Y.-M. Kuo, R. A. Henry, and A. J. Andrews. Measuring specificity in multi-substrate/product systems as a tool to investigate selectivity in vivo. *Biochimica et biophysica acta*, 1864(1):70–6, 2016.

[412] J.-P. Goddard and J.-L. Reymond. Enzyme assays for high-throughput screening. *Current Opinion in Biotechnology*, 15(4):314–322, 2004.

[413] I. Arisi, A. Cattaneo, and V. Rosato. Parameter estimate of signal transduction pathways. *BMC Neuroscience*, 7(SUPPL. 1):S6, 2006.

[414] T. D. Pollard. A guide to simple and informative binding assays. *Molecular biology of the cell*, 21(23):4061–7, 2010.

[415] M. Schliemann-Bullinger, D. Fey, T. Bastogne, R. Findeisen, P. Scheurich, and E. Bullinger. The experimental side of parameter estimation. In L. Geris and D. Gomez-Cabrero, editors, *Uncertainty in Biology: A Computational Modeling Approach*, pages 127–154. Springer International Publishing, Cham, 2016.

[416] G. Cedersund, O. Samuelsson, G. Ball, J. Tegnér, and D. Gomez-Cabrero. Optimization in biology parameter estimation and the associated optimization problem. In L. Geris and D. Gomez-Cabrero, editors, *Uncertainty in Biology: A Computational Modeling Approach*, pages 177–197. Springer International Publishing, Cham, 2016.

[417] A. Nishi, J. A. Bibb, G. L. Snyder, H. Higashi, A. C. Nairn, and P. Greengard. Amplification of dopaminergic signaling by a positive feedback loop. *Proceedings of the National Academy of Sciences*, 97(23):12840–12845, 2000.

[418] A. Nishi, Y. Watanabe, H. Higashi, M. Tanaka, A. C. Nairn, and P. Greengard. Glutamate regulation of DARPP-32 phosphorylation in neostriatal neurons involves activation of multiple signaling cascades. *Proceedings of the National Academy of Sciences of the United States of America*, 102(4):1199–204, 2005.

[419] M. Bouhaddou and M. R. Birtwistle. Kinetic models of biochemical signaling networks. In D. E. Mager and H. H. Kimko, editors, *Systems Pharmacology and Pharmacodynamics*, pages 105–135. Springer International Publishing, Cham, 2016.

[420] M. F. Ciaccio, J. P. Wagner, C.-P. Chuu, D. A. Lauffenburger, and R. B. Jones. Systems analysis of EGF receptor signaling dynamics with microwestern arrays. *Nature Methods*, 7(2):148–155, 2010.

[421] C. Terfve and J. Saez-Rodriguez. Modeling signaling networks using high-throughput phospho-proteomics. *Advances in Experimental Medicine and Biology*, 736:19–57, 2012.

[422] P. Picotti, B. Bodenmiller, L. N. Mueller, B. Domon, and R. Aebersold. Full Dynamic Range Proteome Analysis of S. cerevisiae by Targeted Proteomics. *Cell*, 138(4):795–806, 2009.

[423] C. D. Terfve, E. H. Wilkes, P. Casado, P. R. Cutillas, and J. Saez-Rodriguez. Large-scale models of signal propagation in human cells derived from discovery phosphoproteomic data. *Nature Communications*, 6(1):8033, 2015.

[424] C. Terfve, T. Cokelaer, D. Henriques, A. MacNamara, E. Goncalves, M. K. Morris, M. van Iersel, D. A. Lauffenburger, and J. Saez-Rodriguez. CellNOptR: a flexible toolkit to train

protein signaling networks to data using multiple logic formalisms. *BMC systems biology*, 6(1):133, 2012.

[425] M. Ostrowski, L. Paulevé, T. Schaub, A. Siegel, and C. Guziolowski. Boolean network identification from perturbation time series data combining dynamics abstraction and logic programming. *Biosystems*, 149:139–153, 2016.

[426] H. Ouyang, J. Fang, L. Shen, E. R. Dougherty, and W. Liu. Learning restricted Boolean network model by time-series data. *Eurasip Journal on Bioinformatics and Systems Biology*, 2014(1):1–12, 2014.

[427] T. Geiger, A. Wehner, C. Schaab, J. Cox, and M. Mann. Comparative Proteomic Analysis of Eleven Common Cell Lines Reveals Ubiquitous but Varying Expression of Most Proteins. *Molecular & Cellular Proteomics*, 11(3):M111.014050, 2012.

[428] N. A. Kulak, G. Pichler, I. Paron, N. Nagaraj, and M. Mann. Minimal, encapsulated proteomic-sample processing applied to copy-number estimation in eukaryotic cells. *Nature Methods*, 11(3):319–324, 2014.

[429] R. J. Hause, K. K. Leung, J. L. Barkinge, M. F. Ciaccio, C.-p. Chuu, and R. B. Jones. Comprehensive Binary Interaction Mapping of SH2 Domains via Fluorescence Polarization Reveals Novel Functional Diversification of ErbB Receptors. *PLoS ONE*, 7(9):e44471, 2012.

[430] J. V. Olsen, B. Blagoev, F. Gnad, B. Macek, C. Kumar, P. Mortensen, and M. Mann. Global, In Vivo, and Site-Specific Phosphorylation Dynamics in Signaling Networks. *Cell*, 127 (3):635–648, 2006.